*The Annals of Applied Probability*
2019, Vol. 29, No. 2, 851–874
https://doi.org/10.1214/18-AAP1394
© Institute of Mathematical Statistics, 2019

# OPTIMAL MEAN-BASED ALGORITHMS FOR TRACE RECONSTRUCTION[1,2]

BY ANINDYA DE, RYAN O'DONNELL AND ROCCO A. SERVEDIO

*Northwestern University, Carnegie Mellon University and Columbia University*

In the *(deletion-channel) trace reconstruction problem*, there is an unknown $n$-bit *source string* $x$. An algorithm is given access to independent *traces* of $x$, where a trace is formed by deleting each bit of $x$ independently with probability $\delta$. The goal of the algorithm is to recover $x$ exactly (with high probability), while minimizing samples (number of traces) and running time.

Previously, the best known algorithm for the trace reconstruction problem was due to Holenstein et al. [in *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms* 389–398 (2008) ACM]; it uses $\exp(\widetilde{O}(n^{1/2}))$ samples and running time for any fixed $0 < \delta < 1$. It is also what we call a "mean-based algorithm," meaning that it only uses the empirical means of the individual bits of the traces. Holenstein et al. also gave a lower bound, showing that any mean-based algorithm must use at least $n^{\widetilde{\Omega}(\log n)}$ samples.

In this paper, we improve both of these results, obtaining matching upper and lower bounds for mean-based trace reconstruction. For any constant deletion rate $0 < \delta < 1$, we give a mean-based algorithm that uses $\exp(O(n^{1/3}))$ time and traces; we also prove that any mean-based algorithm must use at least $\exp(\Omega(n^{1/3}))$ traces. In fact, we obtain matching upper and lower bounds even for $\delta$ subconstant and $\rho = 1 - \delta$ subconstant: when $(\log^3 n)/n \ll \delta \le 1/2$ the bound is $\exp(-\Theta(\delta n)^{1/3})$, and when $1/\sqrt{n} \ll \rho \le 1/2$ the bound is $\exp(-\Theta(n/\rho)^{1/3})$.

Our proofs involve estimates for the maxima of Littlewood polynomials on complex disks. We show that these techniques can also be used to perform trace reconstruction with random insertions and bit-flips in addition to deletions. We also find a surprising result: for deletion probabilities $\delta > 1/2$, the presence of insertions can actually *help* with trace reconstruction.

## 1. Introduction.

Consider a setting in which a string $x$ of length $n$ over an alphabet $\Sigma$ is passed through a *deletion channel* that independently deletes each coordinate of $x$ with probability $\delta$. The resulting string, of length somewhere be-

Received September 2017; revised April 2018.

[1] Supported by start up grant from Northwestern University, NSF Grant CCF-1618679, CCF-1420349 and CCF-1563155.

[2] Preliminary version of this paper appeared in the 49th ACM Symposium on Theory of Computing, 2017 [6].

*MSC2010 subject classifications.* Primary 62G07, 68Q32; secondary 94A40.

*Key words and phrases.* Trace reconstruction, deletion channel, Littlewood polynomials.

tween 0 and $n$, is referred to as a *trace* of $x$, or as a *received string*; the original string $x$ is referred to as the *source string*. The *trace reconstruction problem* is the task of reconstructing $x$ (with high probability) given access to independent traces of $x$. This is a natural and well-studied problem, dating back to the early 2000s [2, 13, 14], with some combinatorial variants dating even to the early 1970s [10]. However, perhaps surprisingly, much remains to be discovered both about the information-theoretic and algorithmic complexity of this problem. Indeed, in a 2009 survey [17], Section 7, Mitzenmacher wrote that "the study of [trace reconstruction] is still in its infancy."

Before discussing previous work, we briefly explain why one can assume a binary alphabet without loss of generality. In case of a general $\Sigma$, drawing $O(\frac{\log n}{1-\delta})$ traces will with high probability reveal the entire alphabet $\Sigma' \subseteq \Sigma$ of symbols that are present in $x$. For each symbol $\sigma \in \Sigma'$, we may consider the binary string $x|_\sigma$ whose $i$th character is 1 iff $x_i = \sigma$; a trace of $x$ is easily converted into a trace of $x|_\sigma$, so the trace reconstruction problem for $x$ can be solved by solving the binary trace reconstruction problem for each $x|_\sigma$ and combining the results in the obvious way. For this reason, our work (and most previous work) focuses on the case of a binary alphabet.

1.1. *Prior work.* As described in [17], the trace reconstruction problem can arise in several natural domains, including sensor networks and biology. However, the apparent difficulty of the problem means that there is not too much published work, at least on the problem of "worst-case" trace reconstruction problem ("worst-case" in the sense that the source string may be any element of $\{0, 1\}^n$). Because of this, several prior authors have considered an "average-case" version of the problem in which the source string is assumed to be uniformly random over $\{0, 1\}^n$ and the algorithm is required to succeed with high probability over the random draw of the traces and over the uniform random choice of $x$. This average-case problem seems to have first been studied by Batu et al. [2], who showed that a simple efficient algorithm, which they call Bitwise Majority Alignment, succeeds with high probability for sufficiently small deletion rates $\delta = O(1/\log n)$ using only $O(\log n)$ traces. Subsequent work of Kannan and McGregor [11] gave an algorithm for random $x$ that can handle both deletions and insertions [both at rates $O(1/\log^2 n)$] as well as bit-flips (with constant probability bounded away from 1/2) using $O(\log n)$ traces. Viswanathan and Swaminathan [24] sharpened this result by improving the deletion and insertion rates that can be handled to $O(1/\log n)$. Finally, [8] gave a poly($n$)-time, poly($n$)-trace algorithm for random $x$ that succeeds with high probability for any deletion rate $\delta$ that is at most some sufficiently small absolute constant.

Several researchers have considered, from an information-theoretic rather than algorithmic perspective, various reconstruction problems that are closely related to the (worst-case) trace reconstruction problem. Kalashnik [10] showed that any

$n$-bit string is uniquely specified by its $k$*-deck*, which is the multiset of all its length-$k$ subsequences, when $k = \lfloor n/2 \rfloor$; this result was later reproved by Manvel et al. [15]. Scott [22] subsequently showed that $k = (1 + o(1))\sqrt{n \log n}$ suffices for reconstruction from the $k$-deck for any $x$, and simultaneously and independently Krasnikov and Roditty [12] showed that $k = \lfloor \frac{16}{7}\sqrt{n} \rfloor + 5$ suffices. (McGregor et al. observed in [16] that the result of [22] yields an information-theoretic algorithm using $\exp(\tilde{O}(n^{1/2}))$ traces for any deletion rate $\delta \le 1 - O(\sqrt{\log(n)/n})$, but did not discuss the running time of such an algorithm.) On the other side, successively larger $\Omega(\log n)$ lower bounds on the value of $k$ that suffices for reconstruction of an arbitrary $x \in \{0, 1\}^n$ from its $k$-deck were given by Manvel et al. [15] and Choffrut and Karhumäki [5], culminating in a lower bound of $2^{\Omega(\sqrt{\log n})}$ due to Dudík and Schulman [7].

Surprisingly, few algorithms have been given for the worst-case trace reconstruction problem as defined in the first paragraph of this paper. Batu et al. [2] showed that a variation of their Bitwise Majority Alignment algorithm succeeds efficiently using $O(n \log n)$ traces if the deletion rate $\delta$ is quite low, at most $O(1/n^{1/2+\varepsilon})$. Holenstein et al. [8] gave a "mean-based" algorithm (we explain precisely what is meant by such an algorithm later) that runs in time $\exp(\tilde{O}(\sqrt{n}))$ and uses $\exp(\tilde{O}(\sqrt{n}))$ traces for any deletion rate $\delta$ that is bounded away from 1 by a constant; this is the prior work that is most relevant to our main positive result. Holenstein et al. [8] also gave a lower bound showing that for any $\delta$ bounded away from 0 by a constant, at least $n^{\Omega(\frac{\log n}{\log \log n})}$ traces are required for any mean-based algorithm. Since the result of [8], several researchers (such as [19]) have raised the question of finding (potentially inefficient) algorithms which have a better sample complexity; however, no progress had been made until this work.

One may also ask for trace reconstruction for more general channels, such as those that allow deletions, insertions and bit-flips which are correlated. The only work we are aware of along these lines is that of Andoni et al. [1], which gives results for trace reconstruction for *average-case* words in the presence of insertions, deletions and substitutions on a tree.

### 1.2. *Our results.*

THEOREM 1.1 (Deletion channel positive result). *There is an algorithm for the trace reconstruction problem which, for any constant* $0 < \delta < 1$, *uses* $\exp(O(n^{1/3}))$ *traces and running time.*

Theorem 1.1 significantly improves the running time and sample complexity of the [8] algorithm, which is $\exp(\tilde{O}(n^{1/2}))$ for fixed constant $\delta$. Furthermore, we can actually extend Theorem 1.1 to the case of $\delta = o(1)$ or $\delta = 1 - o(1)$; see Theorem 1.3 below.

The algorithm of Theorem 1.1 is a "mean-based" algorithm, meaning that it uses only the empirical mean of the trace vectors it receives. We prove an essentially matching lower bound for such algorithms.

THEOREM 1.2 (Deletion channel negative result). *For any constant* $0 < \delta < 1$, *every mean-based algorithm must use at least* $\exp(\Omega(n^{1/3}))$ *traces*.

As mentioned, we can also treat $\delta = o(1)$ and $\delta = 1 - o(1)$.

THEOREM 1.3 (Deletion channel general matching bounds). *The matching bounds in Theorems 1.1 and 1.2 extend as follows*: *For* $\Omega(\log^3 n)/n \le \delta \le 1/2$, *the matching bound is* $\exp(\Theta(\delta n)^{1/3})$ [*and for any smaller* $\delta$ *we have a* $\operatorname{poly}(n)$ *upper bound*]. *Writing* $\rho = 1 - \delta$ *for the "retention" probability, for* $\Omega(1/n^{1/2}) \le \rho \le 1/2$ *the matching bound is* $\exp(\Theta(n/\rho)^{1/3})$.

For simplicity in the main portion of the paper, we consider only the deletion channel and prove the above results. In the Appendix, we consider a more general channel that allows for deletions, insertions and bit-flips, and prove the following result, which extends Theorem 1.1 to that more general channel and includes Theorem 1.1 as a special case.

THEOREM 1.4 (General channel positive result). *Let* $\mathcal{C}$ *be the general channel described in Section A.1 with deletion probability* $\delta = 1 - \rho$, *insertion probability* $\sigma$ *and bit-flip probability* $\gamma/2$. *Define*

$$r = \frac{\rho + \delta\sigma}{1 + \sigma}.$$

*Then there is an algorithm for* $\mathcal{C}$*-channel trace reconstruction using samples and running time bounded by*

$$\operatorname{poly}\left(\frac{1}{1-\delta}, \frac{1}{1-\sigma}, \frac{1}{1-\gamma}\right) \times \begin{cases} \exp\bigl(O(n/r)^{1/3}\bigr) & \text{if } C/n^{1/2} \le r \le 1/2, \\ \exp\bigl(O((1-r)n)^{1/3}\bigr) & \text{if } O(\log^3 n)/n \le 1 - r \le 1/2. \end{cases}$$

Since some slight technical and notational unwieldiness is incurred by dealing with the more general channel, we defer the proof of Theorem 1.4 to the Appendix; however, we note here that the main core of the proof is unchanged from the deletion-only case. We additionally note that, as discussed in the Appendix, a curious aspect of the upper bound given by Theorem 1.4 is that having a constant insertion rate can make it possible to perform trace reconstruction in time $\exp(O(n^{1/3}))$ even when the deletion rate is much higher than Theorem 1.3 could handle in the absence of insertions. A possible intuitive explanation for this is that having random insertions could serve to "smooth out" worst-case instances that are problematic for a deletion-only model.

1.3. *Independent and concurrent work.* Simultaneously and independently of the conference publication of this work [6], Fedor Nazarov and Yuval Peres have obtained results [20] that are substantially similar to Theorems 1.1 and 1.2, using very similar techniques. In subsequent recent work, Yuval Peres and Alex Zhai [21] have shown that in the deletion-only setting, for any deletion rate bounded below $1/2$ by a constant there is an algorithm that reconstructs a uniform random source string with high probability using $\exp(O(\log^{1/2} n))$ traces.

Also, Elchanan Mossel has informed us [18] that around 2008, Mark Braverman, Avinatan Hassidim and Elchanan Mossel had independently proven (unpublished) superpolynomial lower bounds for mean-based algorithms.

1.4. *Our techniques.* For simplicity of discussion, we restrict our focus in this section to the question of upper bounding the sample complexity of trace reconstruction for the deletion channel, where every bit gets deleted independently with probability $\delta$. (As discussed above, generalizing the results to channels which also allow for insertions and flips is essentially a technical exercise that does not require substantially new ideas.) As we discuss in Section 3.2, an efficient *algorithm* follows easily from a sample complexity upper bound via the observation that the minimization problem whose solution yields a sample complexity upper bound in fact extends to a slightly larger *convex* set. Given this, one can use convex (in fact, linear) programming to get an algorithmic result. Hence the technical meat of the argument lies in upper bounding the sample complexity.

The key enabling idea for our work is to take an analytic view on the combinatorial process defined by the deletion channel. More precisely, consider two distinct strings $x, x' \in \{-1, 1\}^n$. A necessary (and sufficient) condition to upper bound the sample complexity of trace reconstruction is to lower bound the statistical distance between the two distributions of traces of $x$ versus $x'$ [let us write $\mathcal{C}(x)$ and $\mathcal{C}(x')$ to denote these two distributions]. Since analyzing the statistical distance $d_{\mathrm{TV}}(\mathcal{C}(x), \mathcal{C}(x'))$ between the distributions $\mathcal{C}(x)$ and $\mathcal{C}(x')$ turns out to be a difficult task, we approach it by considering a limited class of statistical tests.

In [8], the authors consider "mean-based" algorithms; such algorithms correspond to statistical tests that only use 1-bit marginals of the distribution of the received string. More precisely, for any $0 \le j \le n-1$, consider the quantities $\mathbf{Pr}_{\boldsymbol{y} \leftarrow \mathcal{C}(x)}[\boldsymbol{y}_j = 1]$ and $\mathbf{Pr}_{\boldsymbol{y}' \leftarrow \mathcal{C}(x')}[\boldsymbol{y}'_j = 1]$. [Here and throughout the paper, the notation "$\mathbf{Pr}_{\boldsymbol{y} \leftarrow \mathcal{C}(x)}$" indicates that the random variable $\boldsymbol{y}$ is distributed according to $\mathcal{C}(x)$.] The difference $|\mathbf{Pr}_{\boldsymbol{y} \leftarrow \mathcal{C}(x)}[\boldsymbol{y}_j = 1] - \mathbf{Pr}_{\boldsymbol{y}' \leftarrow \mathcal{C}(x')}[\boldsymbol{y}'_j = 1]|$ is a lower bound on $d_{\mathrm{TV}}(\mathcal{C}(x), \mathcal{C}(x'))$.

Let us define the vector $\beta_{x,x'} = (\beta_{x,x'}(1), \dots, \beta_{x,x'}(n)) \in [-1, 1]^n$ by

$$\beta_{x,x'}(j) = \mathop{\mathbf{Pr}}_{\boldsymbol{y} \leftarrow \mathcal{C}(x)}[\boldsymbol{y}'_j = 1] - \mathop{\mathbf{Pr}}_{\boldsymbol{y}' \leftarrow \mathcal{C}(x')}[\boldsymbol{y}_j = 1].$$

In this terminology, giving a sample complexity upper bound on mean-based algorithms correspond to showing a lower bound on $\min_{x \neq x' \in \{-1,1\}^n} \|\beta_{x,x'}\|_1$. A central idea in this paper is to analyze $\|\beta_{x,x'}\|_1$ by studying the $Z$-transform of the vector $\beta_{x,x'}$. More precisely, for $z \in \mathbb{C}$, we consider $\widehat{\beta}_{x,x'}(z) := \sum_{j=1}^{n} \beta_{x,x'}(j) \cdot z^{j-1}$. Elementary complex analysis can be used to show the following (see Proposition 3.5):

$$\sup_{|z|=1} \big|\widehat{\beta}_{x,x'}(z)\big| \le \|\beta_{x,x'}\|_1 \le \sqrt{n} \cdot \sup_{|z|=1} \big|\widehat{\beta}_{x,x'}(z)\big|.$$

Thus, for our purposes, it suffices to study $\sup_{|z|=1} |\widehat{\beta}_{x,x'}(z)|$. By analyzing the deletion channel and observing that $\widehat{\beta}_{x,x'}(z)$ is a polynomial in $z$, we are able to characterize this supremum as the supremum of a certain polynomial (induced by $x$ and $x'$) on a certain disk in the complex plane. Thus giving a sample complexity upper bound amounts to lower bounding $\sup_{|z|=1} |\widehat{\beta}_{x,x'}(z)|$ across all polynomials $\widehat{\beta}_{x,x'}$ induced by distinct $x, x' \in \{-1, 1\}^n$ (essentially, across a class of polynomials closely related to *Littlewood polynomials*: those polynomials with all coefficients in $\{-1, 0, 1\}$). The technical heart of our sample complexity upper bound is in establishing such a lower bound. Finally, similar ideas and arguments are used to lower bound the sample complexity of mean-based algorithms by establishing the existence of distinct $x, x' \in \{-1, 1\}^n$ for which $\sup_{|z|=1} |\widehat{\beta}_{x,x'}(z)|$ is small.

## 2. Preliminaries and terminology.

Throughout this paper, we will use two slightly nonstandard notational conventions. Bits will be written as $\{-1, 1\}$ rather than $\{0, 1\}$, and strings will be indexed starting from 0 rather than 1. Thus the *source string* will be denoted $x = (x_0, x_1, \dots, x_{n-1}) \in \{-1, 1\}^n$; this is the unknown string that the reconstruction algorithm is trying to recover.

We will write $\mathcal{C}$ for the *channel* through which $x$ is transmitted. In the main body of the paper, our main focus will be on the *deletion channel* $\mathcal{C} = \mathrm{Del}_\delta$, in which each bit of $x$ is independently deleted with probability $\delta < 1$. We will also often consider $\rho = 1 - \delta > 0$, the retention probability of each coordinate. In the Appendix, we will see that a more general channel that also involves *insertions* and *bit-flips* can be handled in a similar way.

We will use boldface to denote random variables. We typically write $\boldsymbol{y} \leftarrow \mathcal{C}(x)$ to denote that $\boldsymbol{y} = (\boldsymbol{y}_0, \boldsymbol{y}_1, \dots, \boldsymbol{y}_{\boldsymbol{n}-1})$ is a random *trace* (or *received string* or *sample*), obtained by passing $x$ through the channel $\mathcal{C}$. Notice the slight inconvenience that the length of $\boldsymbol{y}$ is a random variable (for the deletion channel this length is always between 0 and $n$); we denote this length by $\boldsymbol{n}$.

We define a *trace reconstruction algorithm for channel* $\mathcal{C}$ to be an algorithm with the following property: for any unknown source string $x \in \{-1, 1\}^n$, when given access to independent strings $\boldsymbol{y}^{(1)}, \boldsymbol{y}^{(2)}, \dots$ each distributed according to $\mathcal{C}(x)$, it outputs $x$ with probability at least (say) 99%. The *sample complexity* of the trace reconstruction algorithm is the number of draws from $\mathcal{C}(x)$ that it uses [in the worst case across all $x \in \{-1, 1\}^n$ and all draws from $\mathcal{C}(x)$]. We are also interested in the algorithm's (worst-case) running time.

As mentioned earlier, we will use basic complex analysis. The following notation will be useful.

NOTATION 2.1. We write $D_r(c)$ for the closed complex disk of radius $r$ centered at $c$; that is, $\{z \in \mathbb{C} : |z - c| \leq r\}$. We write $\partial D_r(c)$ for the boundary of this disk; thus, for example, $\partial D_1(0) = \{z \in \mathbb{C} : |z| = 1\}$ is the complex unit circle.

## 3. Mean traces.

We now come to a key definition, that of the *mean trace*. For now, we restrict our focus to $\mathcal{C}$ being the deletion channel $\mathrm{Del}_\delta$ (we consider a more general channel in the Appendix).

Although a random trace $\boldsymbol{y} \leftarrow \mathrm{Del}_\delta(x)$ does not have a fixed length, we can simply define the mean trace of a source string $x \in \{-1, 1\}^n$ to be

$$\mu_{\mathrm{Del}_\delta}(x) = \mathop{\mathbf{E}}_{\boldsymbol{y} \leftarrow \mathrm{Del}_\delta(x)} [\boldsymbol{y}'] \in [-1, 1]^n, \tag{3.1}$$

where $\boldsymbol{y}'$ is $\boldsymbol{y}$ padded with zeros so as to be of length exactly $n$. Here, "0" has a natural interpretation as a "uniformly random bit" [indeed, a trace reconstruction algorithm could always pad deletion-channel traces with random bits by itself, and this would not change the definition of the mean trace $\mu_{\mathrm{Del}_\delta}(x)$].

The following is immediate.

PROPOSITION 3.1. *Viewing the domain of* $\mu_{\mathrm{Del}_\delta}$ *as the real vector space* $\mathbb{R}^n$, $\mu_{\mathrm{Del}_\delta}(x)$ *is a* (*real-*)*linear function of* $x$; *that is*, *each* $\mu_{\mathrm{Del}_\delta}(x)_j$ *can be written as* $\sum_i a_{i,j} x_i$ *for some constants* $a_{i,j} \in \mathbb{R}$.

### 3.1. *The mean-based* (*deletion-channel*) *trace reconstruction model.*

One of the most basic things that a trace reconstruction algorithm can do is calculate an empirical estimate of the mean trace. A simple Chernoff/union bound shows that, with $\mathrm{poly}(n/\epsilon)$ samples and time, an algorithm can compute an estimator $\widehat{\mu}_{\mathrm{Del}_\delta}(x) \in [-1, 1]^n$ satisfying $\|\widehat{\mu}_{\mathrm{Del}_\delta}(x) - \mu_{\mathrm{Del}_\delta}(x)\|_1 \leq \epsilon$ with very high probability. The algorithm might then proceed to base its reconstruction solely on $\widehat{\mu}_{\mathrm{Del}_\delta}(x)$, without relying on further traces. We call such algorithms "mean-based trace reconstruction algorithms" (Holenstein et al. [8] called them algorithms based on "summary statistics"). We give a formal definition.

DEFINITION 3.2. An algorithm in the *mean-based (deletion-channel) trace reconstruction model* works as follows. Given an unknown source string $x \in \{-1, 1\}^n$, the algorithm first specifies a parameter $T \in \mathbb{N}$. The algorithm is then given an estimate $\widehat{\mu}_{\mathrm{Del}_\delta}(x) \in [-1, +1]^n$ of the mean trace satisfying

$$\|\widehat{\mu}_{\mathrm{Del}_\delta}(x) - \mu_{\mathrm{Del}_\delta}(x)\|_1 \leq 1/T. \tag{3.2}$$

We define the "cost" of this portion of the algorithm to be $T$. Having been given $\widehat{\mu}_{\mathrm{Del}_\delta}(x)$, the algorithm has no further access to $x$, but may do further "post-processing" computation involving $\widehat{\mu}_{\mathrm{Del}_\delta}(x)$. The algorithm should end by outputting $x$.

From the above discussion, we see that an algorithm in the mean-based trace reconstruction model with cost $T_1$ and postprocessing time $T_2$ may be converted into a normal trace reconstruction algorithm using $\mathrm{poly}(n, T_1)$ samples and $\mathrm{poly}(n, T_1) + T_2$ time.

3.2. *The complexity of mean-based* (*deletion-channel*) *trace reconstruction*. As discussed in [8], the sample complexity of mean-based trace reconstruction is essentially determined by the minimum distance between the mean traces $\mu_{\mathrm{Del}_\delta}(x)$ and $\mu_{\mathrm{Del}_\delta}(x')$ of two distinct source strings $x, x' \in \{-1, 1\}^n$. Furthermore, one can get an upper bound on the *time* complexity of mean-based trace reconstruction if a certain "fractional relaxation" of this minimum mean trace distance is large. We state these observations from [8] here, using slightly different notation.

DEFINITION 3.3. Given $n$ and $0 \le \delta < 1$, we define:

$$\begin{aligned}\epsilon_{\mathrm{Del}_\delta}(n) &:= \min_{\substack{x,x' \in \{-1,1\}^n \\ x \neq x'}} \bigl\| \mu_{\mathrm{Del}_\delta}(x) - \mu_{\mathrm{Del}_\delta}(x') \bigr\|_1 \\ &= 2 \min_{\substack{b \in \{-1,0,+1\}^n \\ b \neq 0}} \bigl\| \mu_{\mathrm{Del}_\delta}(b) \bigr\|_1;\end{aligned}$$

$$\begin{aligned}\epsilon^{\mathrm{frac}}_{\mathrm{Del}_\delta}(n) &:= \min_{0 \le i < n} \min_{\substack{x,x' \in [-1,+1]^n \\ x_j = x'_j \in \{-1,1\}\ \forall j < i \\ x_i = -x'_i \in \{-1,1\}}} \bigl\| \mu_{\mathrm{Del}_\delta}(x) - \mu_{\mathrm{Del}_\delta}(x') \bigr\|_1 \\ &= 2 \min_{d \in [n]} \min_{b \in \{0\}^{d-1} \times \{1\} \times [-1,+1]^{n-d}} \bigl\| \mu_{\mathrm{Del}_\delta}(b) \bigr\|_1.\end{aligned}$$

In both cases, the equality on the right uses Proposition 3.1.

It is easy to see that in the mean-based trace reconstruction model, it is information-theoretically possible for an algorithm to succeed if and only if its cost $T$ exceeds $2/\epsilon_{\mathrm{Del}_\delta}(n)$. Thus characterizing the sample complexity of mean-based trace reconstruction essentially amounts to analyzing $\epsilon_{\mathrm{Del}_\delta}(n)$. For example, to establish our lower bound Theorem 1.2, it suffices to prove that the $\epsilon_{\mathrm{Del}_\delta}(n) \le \exp(-\Omega(n^{1/3}))$ for constant $0 < \delta < 1$.

Furthermore, as observed in [8], given an $\epsilon^{\mathrm{frac}}_{\mathrm{Del}_\delta}(n)/4$-accurate estimate of $\mu_{\mathrm{Del}_\delta}(x)$, as well as the ability to compute the linear function $\mu_{\mathrm{Del}_\delta}(x')$ for any $x' \in [-1, +1]^n$ [or even estimate it to $\epsilon^{\mathrm{frac}}_{\mathrm{Del}_\delta}(n)/4$-accuracy], one can recover $x$ exactly in $\mathrm{poly}(n, \log(1/\epsilon^{\mathrm{frac}}_{\mathrm{Del}_\delta}(n)))$ time by solving a sequence of $n$ linear programs.[3] Thus to establish our Theorem 1.1, it suffices to prove that $\epsilon^{\mathrm{frac}}_{\mathrm{Del}_\delta}(n) \ge \exp(-O(n^{1/3}))$ for constant $0 < \delta < 1$.

[3] If the algorithm "knows" $\delta$, it can efficiently compute $\mu_{\mathrm{Del}_\delta}(x')$ exactly. But even if it doesn't "know" $\delta$, it can estimate $\delta$ to sufficient accuracy so that $\mu_{\mathrm{Del}_\delta}(x')$ can be estimated to the necessary accuracy, with no significant algorithmic slowdown.

3.3. *Reduction to complex analysis.* Our next important definition is of a polynomial that encodes the components of $\mu_{\mathcal{C}}(x)$ in its coefficients—kind of a generating function for the channel. We think of its parameter $z$ as a complex number.

DEFINITION 3.4. Given $x \in \{-1, 1\}^n$ and $0 \le \delta < 1$, we define the *deletion-channel polynomial*:

$$P_{\mathrm{Del}_\delta, x}(z) = \sum_{j<n} \mu_{\mathrm{Del}_\delta}(x)_j \cdot z^j,$$

a polynomial of degree less than $n$. We extend this definition to $x \in [-1, +1]^n$ using the linearity of $\mu_{\mathrm{Del}_\delta}$.

We now make the step to elementary complex analysis, by relating the size of a mean trace difference $\mu_{\mathrm{Del}_\delta}(b)$ to the maximum modulus of $P_{\mathrm{Del}_\delta, b}(z)$ on the unit complex circle (or equivalently, the unit complex disk, by the maximum modulus principle).

PROPOSITION 3.5. *For any $b \in [-1, 1]^n$, we have*

$$\max_{z \in \partial D_1(0)} \left| P_{\mathrm{Del}_\delta, b}(z) \right| \le \left\| \mu_{\mathrm{Del}_\delta}(b) \right\|_1 \le \sqrt{n} \max_{z \in \partial D_1(0)} \left| P_{\mathrm{Del}_\delta, b}(z) \right|.$$

PROOF. Recall that $\mu_{\mathrm{Del}_\delta}(b)$ is the length-$n$ vector of coefficients for the polynomial $P_{\mathrm{Del}_\delta, b}(z)$. The lower bound above is immediate from the triangle inequality. For the upper bound, we use

$$\begin{aligned}\left\| \mu_{\mathrm{Del}_\delta}(b) \right\|_1^2 &\le n \left\| \mu_{\mathrm{Del}_\delta}(b) \right\|_2^2 \\ &= n \operatorname*{avg}_{z \in \partial D_1(0)} \left| P_{\mathrm{Del}_\delta, b}(z) \right|^2 \\ &\le n \Big( \max_{z \in \partial D_1(0)} \left| P_{\mathrm{Del}_\delta, b}(z) \right| \Big)^2.\end{aligned}$$

Here, the first inequality is Cauchy–Schwarz, the equality is an elementary fact about complex polynomials (or Fourier series), and the final inequality is obvious. □

Let us reconsider Definition 3.3. As a factor of $\sqrt{n}$ is negligible compared to the bounds, we will prove [which are of the shape $\exp(-\Theta(n^{1/3}))$], we may as well analyze $\max_{z \in \partial D_1(0)} |P_{\mathrm{Del}_\delta, b}(z)|$ rather than $\|\mu_{\mathrm{Del}_\delta}(b)\|_1$ in the definition of $\epsilon_{\mathrm{Del}_\delta}(n)$ and $\epsilon^{\mathrm{frac}}_{\mathrm{Del}_\delta}(n)$. We therefore take a closer look at the deletion-channel polynomial.

**4. The deletion-channel polynomial.** In this section, we compute the deletion-channel polynomial. When the deletion channel is applied to some source string $x$, each bit $x_i$ is either deleted with probability $\delta$ or else is transmitted at some position $j \le i$ in the received string $\boldsymbol{y}$. Let us introduce (nonindependent) random variables $\boldsymbol{J}_0, \dots, \boldsymbol{J}_{n-1}$, where $\boldsymbol{J}_i = \perp$ if $x_i$ is deleted and otherwise $\boldsymbol{J}_i$ is the position in $\boldsymbol{y}$ at which $x_i$ is transmitted. We thus have

$$\begin{aligned} P_{\mathrm{Del}_\delta, x}(z) &= \sum_{j<n} \mathop{\mathbf{E}}_{\boldsymbol{y} \leftarrow \mathcal{C}(x)} [\boldsymbol{y}_j] \cdot z^j = \sum_{j<n} z^j \cdot \sum_{i<n} \mathbf{Pr}[\boldsymbol{J}_i = j] x_i \\ &= \sum_{i<n} x_i \cdot \sum_{j<n} \mathbf{Pr}[\boldsymbol{J}_i = j] z^j = \sum_{i<n} x_i \cdot \mathbf{E}[z^{\boldsymbol{J}_i} \cdot \mathbf{1}[\boldsymbol{J}_i \ne \perp]]. \end{aligned}$$

Observing that $\mathbf{Pr}[\boldsymbol{J}_i \ne \perp]$ equals the retention probability $\rho = 1 - \delta$, if we define the conditional random variable

$$\widetilde{\boldsymbol{J}}_i = (\boldsymbol{J}_i \mid \boldsymbol{J}_i \ne \perp)$$

(so $\widetilde{\boldsymbol{J}}_i$ is an $\mathbb{N}$-valued random variable), then we have

$$P_{\mathrm{Del}_\delta, x}(z) = \rho \sum_{i<n} x_i \cdot \mathbf{E}[z^{\widetilde{\boldsymbol{J}}_i}]. \tag{4.1}$$

Observing that $\widetilde{\boldsymbol{J}}_i$ is distributed as Binomial$(i, \rho)$, and letting $\boldsymbol{B}_1, \dots, \boldsymbol{B}_i$ denote independent Bernoulli random variables with "success" probability $\rho$, we easily compute

$$\mathbf{E}[z^{\widetilde{\boldsymbol{J}}_i}] = \mathbf{E}[z^{\boldsymbol{B}_1 + \cdots + \boldsymbol{B}_i}] = \mathbf{E}[z^{\boldsymbol{B}_1}]^i = ((1-\rho) + \rho z)^i.$$

Denoting

$$w = 1 - \rho + \rho z,$$

we conclude that

$$P_{\mathrm{Del}_\delta, x}(z) = \rho \sum_{i<n} x_i w^i.$$

As $z$ ranges over the unit circle $\partial D_1(0)$, $w$ ranges over the radius-$\rho$ circle $\partial D_\rho(1-\rho)$. Recalling Definition 3.3 and Proposition 3.5, we are led to consider the following two quantities for $0 < \rho < 1$ [note that by the maximum modulus principle, these quantities are unchanged whether the max is over $D_\rho(1-\rho)$ or $\partial D_\rho(1-\rho)$]:

$$\begin{aligned} &\kappa_{\mathrm{Littlewood}}(\rho, n) \\ &\quad = \min\left\{ \max_{w \in D_\rho(1-\rho)} |P(w)| : P(w) = \sum_{i=0}^{n-1} b_i w^i, b_i \in \{0, \pm 1\} \text{ not all } 0 \right\}, \end{aligned}$$

$$\begin{aligned}\kappa^{\mathrm{frac}}_{\mathrm{bounded}}&(\rho, d)\\ &= \min\Biggl\{\max_{w \in D_\rho(1-\rho)} \bigl|P(w)\bigr| : P(w) = w^d\\ &\quad + \sum_{j=d+1}^{N} b_j w^j, N \geq d, b_i \in D_1(0)\Biggr\}.\end{aligned}$$

Observe that both $\kappa_{\mathrm{Littlewood}}(\rho, n)$ and $\kappa^{\mathrm{frac}}_{\mathrm{bounded}}(\rho, d)$ are nondecreasing functions of $0 < \rho < 1$. It's also easy to see that both are nonincreasing functions of their second argument for all $0 < \rho < 1$ [for $\kappa^{\mathrm{frac}}_{\mathrm{bounded}}(\rho, d)$, consider replacing $P(w)$ by $wP(w)$] and observe that $|wP(w)| \leq |P(w)|$ for all $w \in D_\rho(1-\rho)$. It thus follows that

$$\kappa^{\mathrm{frac}}_{\mathrm{bounded}}(\rho, d) \leq \kappa_{\mathrm{Littlewood}}(\rho, n).$$

Our main technical theorems are the following.

THEOREM 4.1. *There is a universal constant $C \geq 1$ such that*

$$\begin{aligned} &\textit{for } 1/d \leq \delta \leq 1/2, && \kappa^{\mathrm{frac}}_{\mathrm{bounded}}(1-\delta, d) \geq \exp\bigl(-C(\delta d)^{1/3}\bigr);\\ &\textit{for } 1/d^{1/2} \leq \rho \leq 1/2, && \kappa^{\mathrm{frac}}_{\mathrm{bounded}}(\rho, d) \geq \exp\bigl(-C(d/\rho)^{1/3}\bigr).\end{aligned}$$

THEOREM 4.2. *There is a universal constant $C \geq 1$ such that*

$$\begin{aligned} &\textit{for } C(\log^3 n)/n \leq \delta \leq 1/2, && \kappa_{\mathrm{Littlewood}}(1-\delta, n) \leq \exp\bigl(-\Omega(\delta n)^{1/3}\bigr);\\ &\textit{for } C/n^{1/2} \leq \rho \leq 1/2, && \kappa_{\mathrm{Littlewood}}(\rho, n) \leq \exp\bigl(-\Omega(n/\rho)^{1/3}\bigr).\end{aligned}$$

By Definition 3.3, Proposition 3.5 and the discussion at the end of Section 3.2, we have that Theorem 4.2 implies both Theorem 1.2 and the more general sample complexity lower bound in Theorem 1.3. Regarding the algorithmic upper bounds in Theorems 1.1 and 1.3, again from Definition 3.3 and Proposition 3.5 we get that

$$\begin{aligned}\epsilon^{\mathrm{frac}}_{\mathrm{Del}_\delta}(n) &\geq 2\rho \cdot \min_{0 \leq d < n}\Biggl\{\max_{w \in D_\rho(1-\rho)} \bigl|P(w)\bigr| : P(w) = w^d\\ &\quad + \sum_{i=d+1}^{n-1} b_i w^i, b_i \in [-1, +1]\Biggr\}\\ &\geq 2\rho \cdot \min_{0 \leq d < n} \kappa^{\mathrm{frac}}_{\mathrm{bounded}}(\rho, d) \geq 2\rho \cdot \kappa^{\mathrm{frac}}_{\mathrm{bounded}}(\rho, n).\end{aligned}$$

Thus the upper bounds Theorems 1.1 and 1.3 likewise follow from Theorem 4.1 and the discussion at the end of Section 3.2. [Note that if $\delta \leq O(\log^3 n)/n$, we can always pay the bound for the larger value $\delta = \Theta(\log^3 n)/n$, which is $\mathrm{poly}(n)$.]

## 5. Proof of Theorem 4.1.

We will need the following:

THEOREM 5.1 ([3], Corollary 3.2, $M = 1$ case). *Let $Q(w)$ be a polynomial with constant coefficient 1 and all other coefficients bounded by 1 in modulus. Fix any $0 < \theta \leq \pi$, and let $A$ be the arc $\{e^{it} : -\theta \leq t \leq \theta\}$. Then $\sup_{w \in A} |Q(w)| \geq \exp(-C_1/\theta)$ for some universal constant $C_1$.*

We remark that for any $0 < r < 1$, Theorem 5.1 holds for the arc $A = \{re^{it} : -\theta \leq t \leq \theta\}$ with no change in the constant $C_1$. This is immediate by applying the theorem to $\widetilde{Q}(w) = Q(rw)$.

PROOF OF THEOREM 4.1. Fix $d \geq 2$ (else the hypotheses are vacuous) and $\delta, \rho \in (0, 1)$ with $\delta + \rho = 1$. We call Case I when $1/d \leq \delta < 1/2$, and we call Case II when $1/d^{1/2} \leq \rho \leq 1/2$. Select

$$\theta = \begin{cases} \dfrac{1}{2(\delta d)^{1/3}} & \text{in Case I,} \\ \left(\dfrac{\rho}{d}\right)^{1/3} & \text{in Case II.} \end{cases}$$

In Case I, we have $\theta \leq 1/2$, and in Case II we have $\theta \leq \rho \leq 1/2$.

Let $P(w) = w^d \cdot Q(w)$, where $Q(w)$ is a polynomial with constant coefficient 1 and all other coefficients bounded by 1 in modulus. We need to show

$$\max_{w \in D_\rho(\delta)} |P(w)| \geq \begin{cases} \exp(-C(\delta d)^{1/3}) & \text{in Case I,} \\ \exp(-C(d/\rho)^{1/3}) & \text{in Case II.} \end{cases} \tag{5.1}$$

In Case I, the ray $\{re^{i\theta} : r > 0\}$ intersects $\partial D_\rho(\delta)$ at a unique point, call it $w_0$. In Case II, the same ray intersects $\partial D_\rho(\delta)$ twice (this uses $\theta \leq \rho$); call the point of larger modulus $w_0$. In either case, consider the triangle formed in the complex plane by the points 0, $\delta$, and $w_0$; it has some acute angle $\alpha$ at $w_0$ and an angle of $\theta$ at 0. By the law of sines,

$$\begin{aligned} \frac{\rho}{\sin\theta} &= \frac{\delta}{\sin\alpha} \\ &= \frac{|w_0|}{\sin(\pi - \theta - \alpha)} \\ &= \frac{|w_0|}{\sin(\theta + \alpha)} \\ &= \frac{|w_0|}{\sin\theta\cos\alpha + \sin\alpha\cos\theta}, \end{aligned}$$

which implies that

$$\begin{aligned}|w_0| &= \delta\cos\theta + \rho\cos\alpha \\ &= \delta\cos\theta + \rho\sqrt{1-\left(\frac{\delta}{\rho}\right)^2\sin^2\theta} \\ &\geq \delta(1-\theta^2) + \rho\left(1-\left(\frac{\delta}{\rho}\right)^2\theta^2\right) \\ &= 1-\frac{\delta}{\rho}\theta^2.\end{aligned}$$

(Note that in Case II, while we have $\frac{\delta}{\rho} \geq 1$, we also have $\theta \leq \rho$ so the quantity inside the square root above is indeed nonnegative.) Writing $r_0 = |w_0|$, Theorem 5.1 (and the subsequent remark) implies that

$$\max_{w\in A}\left|Q(w)\right| \geq \exp(-C_1/\theta) \qquad \text{for } A = \left\{r_0 e^{it} : -\theta \leq t \leq \theta\right\} \subset D_\rho(\delta). \tag{5.2}$$

Thus

$$\begin{aligned}\max_{w\in D_\rho(\delta)}\left|P(w)\right| &\geq \max_{w\in A}\left|P(w)\right| \\ &\geq r_0^d \cdot \exp(-C_1/\theta) \\ &\geq \left(1-(\delta/\rho)\theta^2\right)^d \cdot \exp(-C_1/\theta) \\ &\geq \exp\left(-2(\delta/\rho)\theta^2 d - C_1/\theta\right)\end{aligned}$$

(the last inequality again using $\theta \leq \rho$ in Case II). Substituting in the value of $\theta$ yields (5.1). □

5.1. *An improved version.* Although we do not need it for our application, we can actually provide a stronger version of the results in the previous section that is also self-contained, that is, it does not rely on Borwein and Erdélyi's theorem 5.1. We used that theorem to establish (5.2); but more strongly than (5.2), we can show there exists an arc $A \subset D_\rho(\delta)$ such that

$$\mathrm{GM}_{w\in A}\left|Q(w)\right| \geq \exp\left(-O(1/\theta)\right),$$

where the left-hand side here denotes the *geometric mean* of $|Q|$ along $A$. (Of course, this is at most the max of $|Q|$ along $A$.) To keep the parameters simpler, we will assume $\rho \leq 1/3$ (this is the more interesting parameter regime anyway, and it is sufficient to yield our Theorem 1.1). Our alternate arc $A$ will be

$$A = \left\{1/3 + re^{it} : -\theta \leq t \leq \theta\right\},$$

where $0 < r < 2/3$ is the larger real radius such that $1/3 + re^{\pm i\theta} \in \partial D_\rho(\delta)$. We remark that still $A \subset D_\rho(\delta)$, by virtue of $\theta \leq \rho \leq 1/3$, and it is not hard to show that the the endpoint of $A$; call it $w' = 1/3 + re^{i\theta} \in \partial D_\rho(\delta)$, again satisfies $|w'| \geq 1 - \Omega(\frac{\delta}{\rho}\theta^2)$. Thus instead of using Theorem 5.1 as a black box, we could have completed our proof of Theorem 4.1 using the following.

THEOREM 5.2. *Let $Q(w)$ be a polynomial with constant coefficient* 1 *and all other coefficients in $D_1(0)$. Fix any $0 < \theta \leq \pi$, $0 \leq r \leq 2/3$, and let $A$ be the arc $\{1/3 + re^{it} : -\theta \leq t \leq \theta\}$. Then* $\mathrm{GM}_{w\in A}(|Q(w)|) \geq \frac{9}{18^{\pi/\theta}}$.

Our proof will require one standard fact from the theory of "Mahler measures."

FACT 5.3. *Let $Q$ be a complex polynomial and let $\mathcal{O}$ be a circle in the complex plane with center $c$. Then* $\mathrm{GM}_{w\in\mathcal{O}}(|Q(w)|) \geq |Q(c)|$.

PROOF. By a linear transformation, we may assume $\mathcal{O}$ is the unit circle $\partial D_1(0)$. Express $Q(w) = a_0 \prod_i (w - \alpha_i)$, where the $\alpha_i$'s are the roots of $Q$. Then $\mathrm{GM}_{w\in\mathcal{O}}(|Q(w)|)$—known as $Q$'s *Mahler measure* (see, e.g., [23])—is exactly equal to $|a_0| \prod_{i\in I} |\alpha_i|$, where $I = \{i : |\alpha_i| \geq 1\}$. [Since $\mathrm{GM}_{w\in\mathcal{O}}(|\cdot|)$ is multiplicative, this statement follows immediately from the elementary fact that $\mathrm{GM}_{w\in\mathcal{O}}(|w - \alpha|) = \max\{|\alpha|, 1\}$.] But clearly we have $|a_0| \prod_{i\in I} |\alpha_i| \geq |a_0| \prod_i |\alpha_i| = |Q(0)|$. □

We can now establish Theorem 5.2:

PROOF OF THEOREM 5.2. Using the bounds on $Q$'s coefficients, we have

$$|Q(w)| \leq 1 + |w| + |w|^2 + \cdots = \frac{1}{1 - |w|} \qquad \text{for } w \in D_1(0); \tag{5.3}$$

$$|Q(1/3)| \geq 1 - |1/3| - |1/3|^2 - \cdots = 1/2. \tag{5.4}$$

Let us apply Fact 5.3 with $\mathcal{O} = \partial D_r(1/3) \supset A$, writing $A'$ for the complementary arc to $A$ in $\mathcal{O}$. We get

$$1/2 \leq \mathrm{GM}_{w\in\mathcal{O}}(|Q(w)|) = \mathrm{GM}_{w\in A}(|Q(w)|)^{\theta/\pi} \cdot \mathrm{GM}_{w\in A'}(|Q(w)|)^{1-\theta/\pi}. \tag{5.5}$$

And by (5.3) we have

$$\begin{aligned}\mathop{\mathrm{GM}}_{w\in A'}(|Q(w)|) &\leq \mathop{\mathrm{GM}}_{w\in A'}\left(\frac{1}{1-|w|}\right)\\ &\leq \mathop{\mathrm{GM}}_{w\in\mathcal{O}}\left(\frac{1}{1-|w|}\right) \qquad (5.6)\\ &\leq \mathop{\mathrm{GM}}_{w\in\partial D_{2/3}(1/3)}\left(\frac{1}{1-|w|}\right),\end{aligned}$$

where the second inequality is because the points $w \in A$ only have larger $\frac{1}{1-|w|}$ than the points in $A'$, and the third inequality is because increasing the radius of $\mathcal{O}$ from $r$ to $2/3$ only increases the value of $\frac{1}{1-|w|}$ for points on $\mathcal{O}$. But now for $-\pi < t \leq \pi$, the point $w = 1/3 + (2/3)e^{it} \in D_{2/3}(1/3)$ has $|w|^2 = 1 - \frac{4}{9}(1 - \cos t)$, and hence

$$\frac{1}{1-|w|} = \frac{1}{1 - \sqrt{1 - \frac{4}{9}(1 - \cos t)}} \leq \frac{9}{2(1 - \cos t)}.$$

Thus

$$\begin{aligned}\operatorname*{GM}_{w \in \partial D_{2/3}(1/3)}\left(\frac{1}{1-|w|}\right) &\leq \exp\left(\frac{1}{2\pi}\int_{-\pi}^{\pi} \ln\left(\frac{9}{2(1-\cos t)}\right)dt\right) \\ &= \frac{9}{2}\exp\left(-\frac{1}{2\pi}\int_{-\pi}^{\pi} \ln(1 - \cos t)\,dt\right) = 9,\end{aligned} \tag{5.7}$$

the last integral being known. [One can get a much easier integral, with a slightly worse constant, by lower-bounding $1 - \cos t \geq (2/\pi^2)t^2$.] Combining (5.5), (5.6), (5.7) yields the theorem. □

**6. Proof of Theorem 4.2.** The key ingredient is the following theorem from [4]. (Recall that a *Littlewood polynomial* has all nonzero coefficients either $-1$ or 1.)

THEOREM 6.1 ([4], Theorem 3.3). *For all $k \geq 2$, there is a nonzero Littlewood polynomial $Q_k$ of degree at most $k$ satisfying $|Q_k(t)| \leq \exp(-c_0\sqrt{k})$ for all real $0 \leq t \leq 1$. Here, $c_0 > 0$ is a universal constant.*

By a simple use of the Hadamard three-circle theorem and maximum modulus principle, Borwein and Erdélyi proved in [3] that the polynomials in Theorem 6.1 establish tightness of their Theorem 5.1 (up to the constant $C_1$). We quote a result that appears within their proof.

THEOREM 6.2 ([3], in the first proof of Theorem 3.3 in the "special case," page 11). *There are universal constants $c_1, c_2, c_3 > 0$ such that the following holds: For all $0 < a \leq c_1$, there exists an integer $2 \leq k \leq c_2/a^2$ such that $\max_{w \in D_{6a}(1)} |Q_k(w)| \leq \exp(-c_3/a)$, where $Q_k$ is the nonzero Littlewood polynomial from Theorem 6.1.*

REMARK 6.3. Actually, Borwein and Erdélyi proved this with an elliptical disk $\mathcal{E}_a$ in place of $D_{6a}(1)$, where $\mathcal{E}_a$ has foci at $1 - 8a$ and 1 and major axis $[1 - 14a, 1 + 6a]$. It is easy to see that $D_{6a}(1) \subset \mathcal{E}_a \subset D_{14a}(1)$, so we wrote $D_{6a}(1)$ in Theorem 6.2 for simplicity and because it loses almost nothing.

We can now prove Theorem 4.2. We state here a slightly more precise version.

THEOREM 6.4. *Using the notation $\delta = 1 - \rho$, we have*

$$\kappa_{\text{Littlewood}}(\rho, n) \le \begin{cases} \exp(-\Omega((\delta n)^{1/3})) & \textit{in } \text{Case I:} \quad C(\log^3 n)/n \le \delta \le 1/2, \\ \exp(-\Omega((n/\rho)^{1/3})) & \textit{in } \text{Case II:} \quad C/n^{1/2} \le \rho \le 1/2, \end{cases}$$

*provided $n \ge n_0$. Here, $n_0, C \ge 1$ are universal constants.*

PROOF OF THEOREM 4.2. With $C, C_1 \ge 1$ universal constants to be specified later, select

$$a = \begin{cases} C_1/(\delta n)^{1/3} & \text{in Case I:} \quad C(\log^3 n)/n < \delta \le 1/2, \\ C_1(\rho/n)^{1/3} & \text{in Case II:} \quad 1/n^{1/2} < \rho < 1/2. \end{cases}$$

Assuming $n_0 = n_0(C_1)$ is sufficiently large we get that $a \le c_1$, where $c_1$ is as in Theorem 6.2. Applying that theorem, we obtain

$$\max_{w \in A} |Q_k(w)| \le \exp(-\Omega(1/a)) \qquad (6.1)$$
$$\text{where } A := D_{6a}(1), k \le c_2/a^2 < n/2.$$

Here, the inequality $c_2/a^2 < n/2$ holds in Case I by assuming $n_0 = n_0(C_1, c_2)$ large enough, and in Case II by taking $C_1 = C_1(c_2)$ large enough. Now define

$$P(w) = w^{\lfloor n/2 \rfloor} \cdot Q_k(w),$$
$$\text{a nonzero Littlewood polynomial of degree less than } n.$$

We wish to bound

$$\max_{w \in R} |P(w)|, \qquad R := D_\rho(\delta)$$

by the expression in the theorem statement. For the points $w \in R \cap A$, we are done by (6.1) (and the fact that $|w^{\lfloor n/2 \rfloor}| \le 1$). For the points in $w \in R \setminus A$, we claim that

$$|w|^2 \le 1 - 36\frac{\delta}{\rho}a^2 \le \exp\left(-36\frac{\delta}{\rho}a^2\right) \qquad \forall w \in R \setminus A. \qquad (6.2)$$

Assuming (6.2), we get

$$\begin{aligned} \max_{w \in R \setminus A} |P(w)| &\le \max_{w \in R \setminus A} |w|^{\lfloor n/2 \rfloor} \cdot \max_{w \in R \setminus A} |Q_k(w)| \\ &\le \exp\left(-18\frac{\delta}{\rho}a^2\right)^{\lfloor n/2 \rfloor} \cdot (n/2 + 1) \\ &\le \exp\left(-\Omega\left(n\frac{\delta}{\rho}a^2\right)\right) \cdot (n/2 + 1), \end{aligned}$$

where the factor $n/2+1$ is an upper bound on $|Q_k(w)|$ over all of $D_1(0)$ (recall that $Q_k$ is a Littlewood polynomial of degree less than $n/2$). By inspection, this is sufficient to complete the proof in both Case I and Case II [in Case I we need to assume $C$ large enough to absorb the factor of $(n/2+1)$].

It remains to establish (6.2). For this, we first note that $\rho > 3a$ in both Case I and Case II [Case I is easier to check; for Case II we need to use that $C = C(C_1)$ is sufficiently large]. This in particular means that $R \setminus A \neq \varnothing$. Writing $w_0$ for either of the intersection points of $\partial R$ and $\partial A$, we have $\max_{w \in R\setminus A} |w| \leq |w_0|$. Thus it suffices to upper-bound $|w_0|^2$.

In the complex plane, consider the triangle formed by $\delta$, 1, and $w_0$. Note that $w_0$ has distance $\rho$ from $\delta$ and distance $6a$ from 1. Let $\theta$ denote the triangle's angle at $\delta$. By the cosine law, $(6a)^2 = \rho^2 + \rho^2 - 2\rho^2 \cos\theta$, and hence $\cos\theta = 1 - 18a^2/\rho^2$. Now consider the triangle formed by $\delta$, 0 and $w_0$. Its angle at $\delta$ is $\pi - \theta$ and the adjacent sides have length $\delta$, $\rho$. Thus by the cosine law,

$$\begin{aligned}
|w_0|^2 &= \delta^2 + \rho^2 - 2\delta\rho\cos(\pi - \theta) \\
&= \delta^2 + \rho^2 + 2\delta\rho\cos\theta \\
&= (\delta + \rho)^2 - 36\delta\rho a^2/\rho^2 \\
&= 1 - 36\frac{\delta}{\rho}a^2,
\end{aligned}$$

as needed for (6.2). □

## 7. Conclusions.

A natural direction for future work is to go beyond mean-based algorithms. For example, an efficient algorithm can estimate the covariances of all *pairs* of trace bits. If different source strings lead to sufficiently different trace-covariances, one could potentially get a more efficient trace reconstruction algorithm. Analyzing this strategy is equivalent to analyzing a certain problem concerning the maxima of Littlewood-like polynomials on $\mathbb{C}^2$; however, we could not make any progress on this problem. It would also be interesting to develop lower bound techniques that apply to a broader class of algorithms than just mean-based algorithms.

Finally, we mention that the authors have applied the techniques in this paper (specifically, the technique used in Section 5.1) to several aspects of the population recovery problem. Details will appear in a forthcoming work.

## APPENDIX: RESULTS ON CHANNELS THAT ALLOW INSERTIONS, DELETIONS AND FLIPS

### A.1. Defining the general channel.

We now describe the most general channel $\mathcal{C}$ that we analyze, which we subsequently refer to as "the general channel."

As stated earlier, this channel allows for three different types of corruptions: deletions with probability $\delta$, insertions with probability $\sigma$ and bit-flips with probability $\gamma/2$. We comment that for mean-based algorithms, the presence of bit-flips makes hardly any difference; thus the reader may focus just on the combination of deletions and insertions.

Our definition of this general channel is essentially the same as that of Kannan and McGregor [11]. More precisely, for parameters $\delta, \sigma, \gamma \in [0, 1)$, we define how the channel acts on a single source bit $b \in \{-1, 1\}$:

1. First, the channel performs "insertions"; that is, it repeatedly does the operation "with probability $\sigma$, transmit a uniformly random bit; with probability $1-\sigma$, stop."
2. Having stopped, the channel "deletes" (completes transmission without sending $b$ or $-b$) with probability $\delta$.
3. Otherwise (with probability $1 - \delta$), the channel transmits one more bit: namely, $b$ with probability $1 - \gamma/2$, or $-b$ with probability $\gamma/2$.

As usual, the channel operates on an entire source string $x \in \{-1, 1\}^n$ by operating on its individual bits independently, concatenating the results. That is,

$$\mathcal{C}(x) = \mathcal{C}(x_0)\mathcal{C}(x_1)\cdots\mathcal{C}(x_{n-1}) \in \{-1, 1\}^*.$$

Of course, if we set $\sigma = \gamma = 0$, we get the deletion channel $\mathrm{Del}_\delta$ that was analyzed in the main body of the paper.

An alternative description of the channel's operation on a single bit $x_i$ is as follows:

$$\mathcal{C}(x_i) = \begin{cases} \boldsymbol{w} & \text{with probability}\delta, \\ (\boldsymbol{w}, \boldsymbol{a}) & \text{with probability } (1-\delta)\cdot\gamma, \\ (\boldsymbol{w}, x_i) & \text{with probability } (1-\delta)\cdot(1-\gamma), \end{cases} \tag{A.1}$$

where $\boldsymbol{a} \in \{-1, 1\}$ is a uniformly random bit, and where $\boldsymbol{w} \in \{-1, 1\}^{\boldsymbol{G}}$ is a uniformly random string of $\boldsymbol{G}$ bits, with $\boldsymbol{G}$ in turn being a Geometric random variable of parameter $1 - \sigma$.[4] From this description, one can see that in a received word $\boldsymbol{y} \leftarrow \mathcal{C}(x)$, each received bit either "comes from a properly transmitted source bit $x_i$," or else is uniformly random. [The probability each $x_i$ comes through is $(1-\delta)(1-\gamma)$.] As a consequence, we have that Proposition 3.1 continues to hold for $\mathcal{C}$: for every $j \in \mathbb{N}$, the mean value $\mathbf{E}_{\boldsymbol{y}\leftarrow\mathcal{C}(x)}[\boldsymbol{y}_j]$ is a (real-)linear function of $x$.

Note that when the insertion probability $\sigma$ is positive, the received word $\boldsymbol{y} \leftarrow \mathcal{C}(x)$ does not have an a priori bounded length. This is a minor annoyance that can be handled in several different ways; we choose one way in the next section.

[4] Here, we use the convention that Geometric random variables take values $0, 1, 2, \ldots$ (equal to the number of "failures"); that is, $\mathbf{Pr}[\boldsymbol{G} = t] = \sigma^t(1-\sigma)$ for each $t \geq 0$.

**A.2. Mean traces for the general channel.** We revisit some of our definitions and observations about mean traces from Section 3, in our new context of the general channel. We begin with (3.1), the definition of the mean trace. Since the length of a received word may now be arbitrarily large, the mean trace is now an infinite vector. We deal with this by truncating it at what we call the "effective trace length bound $N$."

DEFINITION A.1. For the general channel $\mathcal{C}$ with insertion probability $0 \leq \sigma < 1$, we define the *effective trace length bound* $N = N(\sigma)$ to be $N = \lceil 10 \times \frac{n+\ln(1/(1-\sigma))}{1-\sigma} \rceil \leq \mathrm{poly}(n, \frac{1}{1-\sigma})$.

DEFINITION A.2. For the general channel $\mathcal{C}$ and a source string $x \in \{-1, 1\}^n$, we define the *idealized mean trace* to be the infinite sequence

$$\mu_{\mathcal{C}}^{\mathrm{ideal}}(x) = \mathop{\mathbf{E}}_{\boldsymbol{y}\leftarrow\mathcal{C}(x)}\bigl[(\boldsymbol{y}, 0, 0, 0, \dots)\bigr] \in [-1, +1]^{\mathbb{N}}.$$

We define just the *mean trace* to be its truncation to length $N$:

$$\mu_{\mathcal{C}}(x) = \bigl(\mu_{\mathcal{C}}^{\mathrm{ideal}}(x)_0, \mu_{\mathcal{C}}^{\mathrm{ideal}}(x)_1, \dots, \mu_{\mathcal{C}}^{\mathrm{ideal}}(x)_{N-1}\bigr) \in [-1, +1]^N.$$

Recalling (A.1), we see that the length $\boldsymbol{n}$ of a received word is stochastically dominated by $(\boldsymbol{G}_1 + 1) + \cdots + (\boldsymbol{G}_n + 1)$, where the $\boldsymbol{G}_i$'s are i.i.d. random variables distributed as Geometric$(1 - \sigma)$. We upper bound this using Janson's bound on the sum of independent Geometric random variables (Theorem 2.1 of [9]), noting that his Geometric random variables count the number of "trials," which aligns precisely with our $(\boldsymbol{G}_i + 1)$'s. His bound gives that $\mathbf{Pr}[\boldsymbol{n} \geq N + j] \leq \exp(-(N + j)(1-\sigma)/2)$ for any $j \geq 0$, and hence we have the following: for any $x \in [-1, 1]^n$,

$$\begin{aligned}
&\|\mu_{\mathcal{C}}(x) - \mu_{\mathcal{C}}^{\mathrm{ideal}}(x)\|_1 \\
&\quad = \sum_{\ell=N}^{\infty} |\mu_{\mathcal{C}}^{\mathrm{ideal}}(x)_\ell| \leq \sum_{\ell=N}^{\infty} \mathbf{Pr}[\boldsymbol{n} \geq \ell] \\
&\quad = \sum_{j=0}^{\infty} \mathbf{Pr}[\boldsymbol{n} \geq N + j] \\
&\quad = \exp\bigl(-N(1-\sigma)/2\bigr) \cdot \frac{1}{1 - \exp(-(1-\sigma)/2)} \\
&\quad < \frac{4\exp(-N(1-\sigma)/2)}{1-\sigma} \\
&\quad \leq 4\exp(-n) \qquad \text{by our choice of } N.
\end{aligned} \tag{A.2}$$

**The mean-based trace reconstruction model for the general channel.** Definition 3.2 has a natural analogue for the general channel: an algorithm in the mean-based general-channel model specifies a cost parameter $T \in \mathbb{N}$ and is given an estimate $\widehat{\mu}_{\mathcal{C}}(x) \in [-1, 1]^N$ of the mean trace satisfying $\|\widehat{\mu}_{\mathcal{C}}(x) - \mu_{\mathcal{C}}(x)\|_1 \le 1/T$. It is clear that an algorithm in the mean-based general-channel trace reconstruction model with cost $T_1$ and postprocessing time $T_2$ may be converted into a normal trace reconstruction algorithm using $\mathrm{poly}(N, T_1) = \mathrm{poly}(n, \frac{1}{1-\sigma}, T_1)$ samples and $\mathrm{poly}(n, \frac{1}{1-\sigma}, T_1) + T_2$ time. Note that since we will be studying algorithms with cost $T \ll 2^n$, by (A.2) there is no real difference between getting an estimate of $\mu_{\mathcal{C}}(x)$ or of $\mu_{\mathcal{C}}^{\mathrm{ideal}}(x)$.

**The complexity of mean-based trace reconstruction for the general channel.** Regarding the complexity of mean-based trace reconstruction, for the general channel we define $\epsilon_{\mathcal{C}}(n)$ and $\epsilon_{\mathcal{C}}^{\mathrm{frac}}(n)$ in the obvious way, replacing each occurrence of the length-$n$ vector $\mu_{\mathrm{Del}_\delta}(\cdot)$ in Definition 3.3 with the length-$N$ vector $\mu_{\mathcal{C}}(\cdot)$. As in Section 3.2, to show that trace reconstruction can be performed under the general channel in time $\mathrm{poly}(N, M) = \mathrm{poly}(n, \frac{1}{1-\sigma}, M)$ it suffices to show that $\epsilon_{\mathcal{C}}^{\mathrm{frac}}(n) \ge 1/M$.[5]

**Reduction to complex analysis for the general channel.** For $x \in \{-1, 1\}^n$ the *general-channel polynomial* is defined entirely analogously to Definition 3.4:

$$P_{\mathcal{C},x}(z) = \sum_{j<N} \mu_{\mathcal{C}}(x)_j \cdot z^j;$$

note that this is a polynomial of degree less than $N$. This definition extends to $x \in [-1, +1]^n$ using the linearity of $\mu_{\mathcal{C}}$. Similarly, we may define the *idealized general-channel "polynomial"* by

$$P_{\mathcal{C},x}^{\mathrm{ideal}}(z) = \sum_{j\in\mathbb{N}} \mu_{\mathcal{C}}^{\mathrm{ideal}}(x)_j \cdot z^j;$$

this will actually be a rational function of $z$.

Entirely analogous to Proposition 3.5, we get that for every $b \in [-1, 1]^n$,

$$\max_{z\in\partial D_1(0)} |P_{\mathcal{C},b}(z)| \le \|\mu_{\mathcal{C}}(b)\|_1 \le \sqrt{N} \max_{z\in\partial D_1(0)} |P_{\mathcal{C},b}(z)|.$$

Similar to Section 3.3, a factor of $\sqrt{N} = \mathrm{poly}(n, \frac{1}{1-\sigma})$ is negligible compared to the bounds we will prove, so it suffices to analyze $\max_{z\in\partial D_1(0)} |P_{\mathcal{C},b}(z)|$ rather than $\|\mu_{\mathcal{C}}(b)\|_1$ in the definitions of $\epsilon_{\mathcal{C}}(n)$ and $\epsilon_{\mathcal{C}}^{\mathrm{frac}}(n)$. Moreover, since by (A.2) we have that $|P_{\mathcal{C},b}^{\mathrm{ideal}}(z) - P_{\mathcal{C},b}(z)| \le 2^{-n}$ for all $b \in [-1, 1]^n$ and all $z \in \partial D_1(0)$, it suffices to analyze $\max_{z\in\partial D_1(0)} |P_{\mathcal{C},b}^{\mathrm{ideal}}(z)|$; we do this in the next subsection.

[5] Again, to carry out the linear-programming algorithm, we can either assume that the channel parameters $\delta$, $\sigma$, $\gamma$ are known to the algorithm, or else they should estimated; we omit the details here.

**A.3. Channel polynomial for general channels.** We now compute the ideal channel polynomial for the general channel defined in Section A.1, using the same technique as in Section 4 and recalling the discussion around the alternative channel description (A.1). As usual, let $\rho = 1 - \delta$. Let $\boldsymbol{J}_i$ be the random variable whose value is $\perp$ if $x_i$ is either deleted (probability $\delta$) or is replaced by a random bit [probability $(1-\delta)\cdot\gamma$], or else is the position $j$ such that coordinate $x_i$ of the source string ends up in coordinate $j$ in the received string $\boldsymbol{y}$. As before, we let $\widetilde{\boldsymbol{J}}_i$ denote the random variable $\boldsymbol{J}_i$ conditioned on not being $\perp$. Since $\mathbf{Pr}[\boldsymbol{J}_i \neq \perp] = (1-\delta)\cdot(1-\gamma)$, a derivation identical to that of (4.1) yields

$$P^{\text{ideal}}_{C,x}(z) = (1-\delta)(1-\gamma)\sum_{i<n} x_i \cdot \mathbf{E}[z^{\widetilde{\boldsymbol{J}}_i}]. \tag{A.3}$$

To compute $\mathbf{E}[z^{\widetilde{\boldsymbol{J}}_i}]$, it is straightforward to see that each coordinate $x_{i'}$ with $i' < i$ independently generates a random number of received positions distributed as $\boldsymbol{G} + \boldsymbol{B}$, where $\boldsymbol{G} \sim \text{Geometric}(1-\sigma)$ and independently $\boldsymbol{B} \sim \text{Bernoulli}(\rho)$. Further, conditioned on $x_i$ not being deleted, $x_i$ generates a number of received positions distributed as $\boldsymbol{G} + 1$, where the final "+1" is for $x_i$ (or $-x_i$) itself. Thus $\widetilde{\boldsymbol{J}}_i$ is distributed as

$$\boldsymbol{G}_0 + \cdots + \boldsymbol{G}_i + \boldsymbol{B}_0 + \cdots + \boldsymbol{B}_{i-1},$$

where the $\boldsymbol{G}_k$'s are independent copies of $\boldsymbol{G}$ and the $\boldsymbol{B}_k$'s are independent copies of $\boldsymbol{B}$. We therefore obtain

$$\mathbf{E}[z^{\widetilde{\boldsymbol{J}}_i}] = \mathbf{E}[z^{\boldsymbol{G}}]^{i+1} \cdot \mathbf{E}[z^{\boldsymbol{B}}]^i = (\mathbf{E}[z^{\boldsymbol{G}}] \cdot \mathbf{E}[z^{\boldsymbol{B}}])^i \cdot \mathbf{E}[z^{\boldsymbol{G}}].$$

Let $F_G(z)$ denote $\mathbf{E}[z^{\boldsymbol{G}}]$ and let $F_B(z)$ denote $\mathbf{E}[z^{\boldsymbol{B}}]$. It is easy to calculate that $F_G(z) = \frac{1-\sigma}{1-\sigma z}$, and we saw earlier that $F_B(z) = (1-\rho) + \rho z = \delta + \rho z$. For brevity, let us write

$$w = F_G(z) F_B(z) = \frac{(1-\sigma)\cdot(\delta+\rho z)}{1-\sigma z},$$

which is a Möbius transformation of $z$. Thus $w$ ranges over a complex circle as $z$ ranges over $\partial D_1(0)$. More specifically, as $z$ ranges over $\partial D_1(0)$ we have that $w$ ranges over $\partial D_r(1-r)$, where

$$r = \frac{\rho + \delta\sigma}{1+\sigma}.$$

Plugging this back into (A.3) using $\mathbf{E}[z^{\widetilde{\boldsymbol{J}}_i}] = F_G(z)\cdot w^i$, we obtain

$$\begin{aligned} P^{\text{ideal}}_{C,x}(z) &= (1-\delta)\cdot(1-\gamma)\cdot F_G(z)\cdot\sum_{i<n} x_i \cdot w^i \\ &= (1-\gamma)\cdot(1-\delta)\cdot\frac{1-\sigma}{1-\sigma z}\cdot\sum_{i<n} x_i\cdot w^i. \end{aligned}$$

We use the bound $|\frac{1-\sigma}{1-\sigma z}| \geq \frac{1-\sigma}{2}$ for $z \in \partial D_1(0)$. Now by the analysis of $\kappa^{\mathrm{frac}}_{\mathrm{bounded}}(r, d)$ given in Section 4 we get the following algorithmic result for general-channel trace reconstruction, which is our most general positive result.

THEOREM 1.4, RESTATED. *Let $\mathcal{C}$ be the general channel described in Section A.1 with deletion probability $\delta = 1 - \rho$, insertion probability $\sigma$, and bit-flip probability $\gamma/2$. Define*

$$r := \frac{\rho + \delta\sigma}{1 + \sigma}.$$

*Then there is an algorithm for $\mathcal{C}$-channel trace reconstruction using samples and running time bounded by*

$$\mathrm{poly}\left(\frac{1}{1-\delta}, \frac{1}{1-\sigma}, \frac{1}{1-\gamma}\right) \cdot \begin{cases} \exp(O(n/r)^{1/3}) & \textit{if } C/n^{1/2} \leq r \leq 1/2, \\ \exp(O((1-r)n)^{1/3}) & \textit{if } O(\log^3 n)/n \leq 1 - r \leq 1/2. \end{cases}$$

Let us make some observations about this result. First, our Theorem 1.1 for the deletion channel is the special case of Theorem 1.4 obtained by setting $\sigma = \gamma = 0$. Next, for fixed $\delta$,

if $\delta \leq 1/2$, $r$ ranges from $1 - \delta$ down to $1/2$ as $\sigma$ ranges from 0 up to 1;

if $\delta \geq 1/2$, $r$ ranges from $1 - \delta$ *up to* $1/2$ as $\sigma$ ranges from 0 up to 1.

The second statement is rather peculiar: it implies that when the deletion rate is high, the ability to perform trace reconstruction actually *improves*, the more insertions there are. Indeed, when we have deletions only, our ability to do trace reconstruction in time $\exp(O(n^{1/3}))$ is limited to retention probability $\rho \geq \Omega(1)$. But as soon as the insertion rate $\sigma$ satisfies $\sigma \geq \Omega(1)$, we can do trace reconstruction in time $\exp(O(n^{1/3}))$ as long as the retention rate $\rho = 1 - \delta$ satisfies $\rho \geq \exp(-O(n^{1/3}))$. While the authors find this result quite counterintuitive, we note that we cannot simulate the insertion plus deletion channel given access to traces from the deletion channel. Thus, one cannot immediately obtain better trace reconstruction bounds for the deletion channel by artificially adding insertions.

**Acknowledgments.** The authors would like to thank the Simons' Foundation for sponsoring the symposium on analysis of Boolean functions where the authors began work on this project. A. De would like to thank Aravindan Vijayaraghavan for useful discussions about this problem.

## REFERENCES

[1] ANDONI, A., DASKALAKIS, C., HASSIDIM, A. and ROCH, S. (2012). Global alignment of molecular sequences via ancestral state reconstruction. *Stochastic Process. Appl.* **122** 3852–3874. MR2971717

[2] BATU, T., KANNAN, S., KHANNA, S. and MCGREGOR, A. (2004). Reconstructing strings from random traces. In *Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms* 910–918. ACM, New York. MR2290981

[3] BORWEIN, P. and ERDÉLYI, T. (1997). Littlewood-type problems on subarcs of the unit circle. *Indiana Univ. Math. J.* **46** 1323–1346. MR1631600

[4] BORWEIN, P., ERDÉLYI, T. and KÓS, G. (1999). Littlewood-type problems on [0, 1]. *Proc. Lond. Math. Soc.* (3) **79** 22–46. MR1687555

[5] CHOFFRUT, C. and KARHUMÄKI, J. (1997). Combinatorics of words. In *Handbook of Formal Languages, Vol.* 1 329–438. Springer, Berlin. MR1469998

[6] DE ANINDYA, A., O'DONNELL, R. and SERVEDIO, R. A. (2017). Optimal mean-based algorithms for trace reconstruction. In *STOC'*17*—Proceedings of the* 49*th Annual ACM SIGACT Symposium on Theory of Computing* 1047–1056. ACM, New York. MR3678250

[7] DUDÍK, M. and SCHULMAN, L. J. (2003). Reconstruction from subsequences. *J. Combin. Theory Ser. A* **103** 337–348. MR1996071

[8] HOLENSTEIN, T., MITZENMACHER, M., PANIGRAHY, R. and WIEDER, U. (2008). Trace reconstruction with constant deletion probability and related results. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms* 389–398. ACM, New York. MR2487606

[9] JANSON, S. (2014). Tail bounds for sums of geometric and exponential variables. Available at http://www2.math.uu.se/~svante/papers/sjN14.pdf.

[10] KALASHNIK, V. V. (1973). Reconstruction of a word from its fragments. *Computational Mathematics and Computer Science* (*Vychislitel'naya Matematika i Vychislitel'naya Tekhnika*), *Kharkov* **4** 56–57.

[11] KANNAN, S. and MCGREGOR, A. (2005). More on reconstructing strings from random traces: Insertions and deletions. In *IEEE International Symposium on Information Theory* 297–301.

[12] KRASIKOV, I. and RODITTY, Y. (1997). On a reconstruction problem for sequences. *J. Combin. Theory Ser. A* **77** 344–348. MR1429086

[13] LEVENSHTEIN, V. I. (2001). Efficient reconstruction of sequences from their subsequences or supersequences. *J. Combin. Theory Ser. A* **93** 310–332. MR1805300

[14] LEVENSHTEIN, V. I. (2001). Efficient reconstruction of sequences. *IEEE Trans. Inform. Theory* **47** 2–22. MR1819952

[15] MANVEL, B., MEYEROWITZ, A., SCHWENK, A., SMITH, K. and STOCKMEYER, P. (1991). Reconstruction of sequences. *Discrete Math.* **94** 209–219. MR1138599

[16] MCGREGOR, A., PRICE, E. and VOROTNIKOVA, S. (2014). Trace reconstruction revisited. In *Algorithms—ESA* 2014. *Lecture Notes in Computer Science* **8737** 689–700. Springer, Heidelberg. MR3253172

[17] MITZENMACHER, M. (2009). A survey of results for deletion channels and related synchronization channels. *Probab. Surv.* **6** 1–33. MR2525669

[18] MOSSEL, E. (2016). Personal communication, October 2016.

[19] MOSSEL, E. (2013). MSRI open problem session. Available at https://www.msri.org/c/document_library/get_file?uuid=4a885484-bcdd-4238-a3da-21c05713034c&groupId=14404.

[20] NAZAROV, F. and PERES, Y. (2017). Trace reconstruction with $\exp(O(n^{1/3}))$ samples. In *STOC'*17*—Proceedings of the* 49*th Annual ACM SIGACT Symposium on Theory of Computing* 1042–1046. ACM, New York. MR3678249

[21] PERES, Y. and ZHAI, A. (2017). Average-case reconstruction for the deletion channel: Subpolynomially many traces suffice. In 58*th Annual IEEE Symposium on Foundations of Computer Science—FOCS* 2017 228–239. IEEE Computer Soc., Los Alamitos, CA. MR3734232

[22] SCOTT, A. D. (1997). Reconstructing sequences. *Discrete Math*. **175** 231–238. MR1475850

[23] SMYTH, C. (2008). The Mahler measure of algebraic numbers: A survey. In *Number Theory and Polynomials*. *London Mathematical Society Lecture Note Series* **352** 322–349. Cambridge Univ. Press, Cambridge. MR2428530

[24] VISWANATHAN, K. and SWAMINATHAN, R. (2008). Improved string reconstruction over insertion-deletion channels. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms* 399–408. ACM, New York. MR2487607

A. DE
ELECTRICAL ENGINEERING
AND COMPUTER SCIENCE
NORTHWESTERN UNIVERSITY
2133 SHERIDAN ROAD
EVANSTON, ILLINOIS 60201
USA
E-MAIL: anindya.de1@northwestern.edu

R. O'DONNELL
SCHOOL OF COMPUTER SCIENCE
CARNEGIE MELLON UNIVERSITY
7213 GATES CENTER
PITTSBURGH, PENNSYLVANIA 15213
USA
E-MAIL: odonnell@cs.cmu.edu

R. SERVEDIO
COMPUTER SCIENCE DEPARTMENT
COLUMBIA UNIVERSITY
500 W. 120TH STREET
NEW YORK, NEW YORK 10027
USA
E-MAIL: rocco@cs.columbia.edu