

Sorting from Noisier Samples

Aviad Rubinfeld* Shai Vardi†

Abstract

We study the problem of constructing an order over a set of elements given noisy samples. We consider two models for generating the noisy samples; in both, the distribution of samples is induced by an unknown state of nature: a permutation ρ . In *Mallow's model*, r permutations π_i are generated independently from ρ , each with probability proportional to $e^{-\beta d_K(\rho, \pi_i)}$, where $d_K(\rho, \pi_i)$ is the Kemeny distance between ρ and π_i - the number of pairs they order differently. In the *noisy comparisons* model, we are given a tournament, generated from ρ as follows: if i is before j in ρ , then with probability $1/2 + \gamma$, the edge between them is oriented from i to j . Both of these problems were studied by Braverman and Mossel [7]; they showed how to construct a maximum-likelihood permutation when the noise parameter (β or γ , respectively) is constant.

In this work, we obtain algorithms that work in the presence of stronger noise ($\beta^2 r = \tilde{\Omega}\left(\frac{1}{\log^2 n}\right)$ or $\gamma = \tilde{\Omega}\left(\frac{1}{\log^{1/6} n}\right)$, respectively). In Mallow's model, our algorithm works for a relaxed solution concept: *likelier than nature*. That is, rather than requiring that our output maximizes the likelihood over the entire domain, we guarantee that the likelihood of our output is, w.h.p., greater than or equal to that of the true state of nature (ρ). An interesting feature of our algorithm is that it handles noise by adding more noise.

1 Introduction

We study the problem of reconstructing a full order over a set of elements from noisy observations. Assume that there is some true underlying ranking of items that we would like to recover. To do so, we ask several experts, and receive from each one a permutation of the items. It is natural to assume that the experts' permutations will be noisy views of the true order. Given these permutations, what can we say about true state of nature? As a second example, consider a basketball league where every team plays every other team, and suppose that there is some true full order of the teams' skills. The result of each match can be thought of as a noisy sample of the teams' relative strengths. In real sports leagues, teams are often ranked by the number of wins, but it is not hard to see that this may not be the best estimate of the true order.

1.1 The models

We consider two models for generating observed data from a permutation: *Mallow's model* captures the first example of experts reporting noisy permutations; the results of the basketball games are modeled as *noisy comparisons*.

*UC Berkeley, Berkeley, CA, USA. E-mail: aviad@eecs.berkeley.edu. This research was supported in part by Microsoft Research PhD Fellowship, as well as NSF grant CCF1408635 and by Templeton Foundation grant 3966. This work was done in part at the Simons Institute for the Theory of Computing.

†Weizmann Institute of Science, Rehovot, Israel. E-mail: shaivardi@gmail.com. This research was supported in part by the I-CORE in Algorithms Postdoctoral Fellowship.

Mallow’s Reconstruction Problem

In *Mallow’s model* [19], there is some (true) full order, ρ , over n elements. The probability of sampling a permutation π is inverse exponential in the Kemeny distance d_K between ρ and π , with some parameter $\beta > 0$:¹

$$\Pr[\pi|\rho] = \frac{1}{Z(\beta, n)} e^{-\beta d_K(\pi, \rho)}, \quad (1)$$

where $Z(\beta, n)$ is the normalization factor, and the *Kemeny distance*² between ρ and π is the number of pairs i, j that π and ρ order differently:

$$d_K(\pi, \rho) = |\{(i, j) : (\pi(i) > \pi(j)) \wedge (\rho(i) < \rho(j))\}|. \quad (2)$$

In *Mallow’s reconstruction problem* (Definition 2.1), we are given n , β , and r permutations, π_1, \dots, π_r , generated independently at random from some permutation ρ according to (1), and our goal is to estimate the original permutation.

Noisy Comparisons

In this model, we are given access to noisy queries of the form “is i better than j ?” In terms of our example from earlier, we are given the winner of a basketball game. Queries are answered independently at random, conditioned on an unknown underlying permutation ρ . In particular, for noise parameter γ , the query on each pair agrees with ρ with probability $\frac{1}{2} + \gamma$. Another way of thinking about the input is as a tournament graph over n vertices where the edges take random directions. Given as input a tournament and a value γ , the *simple noisy signal aggregation problem* (Definition 2.2) is to find the permutation that was most likely to generate this tournament.

1.2 Our Contributions

Our work is a direct extension of the works of Braverman and Mossel [6, 7], who gave algorithms that reconstruct the maximum likelihood permutations on the same models. We borrow heavily from their techniques; however, applying some new ideas allows us to get significantly better noise tolerance.

A relaxed solution concept. We introduce a relaxed solution concept: *likelier than nature*. In settings with large noise-to-samples ratio, the maximum likelihood permutation could be quite far from the true order; in this case, it is supererogatory to pinpoint the maximum likelihood permutation. Instead, we guarantee that our algorithm’s output has likelihood greater than or equal to that of the true order. We argue that this guarantee is at least as useful as finding the exact maximum likelihood permutation: once we are guaranteed that a solution is at least as likely as the true state of nature, finding another such solution is redundant.

Mallow’s model. In Mallow’s model, when the noise-to-sample size ratio is very low ($\min\{\beta^2, 1\} \cdot r = \Omega(\log n)$), the relative order of every pair of elements in the majority of permutations agrees (w.h.p.) with the true order; therefore the latter can be reconstructed exactly by a simple sorting algorithm (e.g., [7, 8]). In the presence of stronger noise³ ($\beta = \Omega(1)$), the maximum likelihood is

¹Smaller values of β correspond to noisier samples.

²Sometimes also called Kendall’s τ , or Inversion Distance.

³Braverman and Mossel’s analysis [7] requires that $r = O(1)$, so their setting is not directly comparable to that of Chierichetti et al. [8]. Our noise tolerance improves over both [7] and [8].

Algorithm	Noise tolerance	Solution concept
Pairwise comparisons [8]	$\min\{\beta^2, 1\} \cdot r = \Omega(\log n)$	True order
Dynamic program [7]	$\beta = \Omega(1)$	Maximum-likelihood
This work	$\beta^2 r = \tilde{\Omega}(\frac{1}{\log^2 n})$	Likelier-than-nature

Table 1: Mallow’s model: a table of the trade-off between solution concepts and noise tolerance. β is the noise parameter and r is the number of samples.

guaranteed to be close to the true order [7]. In this case, one may hope to compute the maximum likelihood order - and indeed this is achieved by Braverman and Mossel [7] via a dynamic program. In the presence of yet stronger noise ($\beta^2 r = \tilde{\Omega}(1/\log^2 n)$), it becomes difficult to argue even about the likelihood maximizer. Nevertheless, we can obtain a permutation that is “likelier than nature” in polynomial time by generalizing the dynamic program from [6, 7].

Theorem 1.1. (Rephrased as Theorem 4.1). *Let π_1, \dots, π_r be permutations on n elements, generated independently according to Mallow’s model with noise parameter β . If $\beta^2 r = \Omega\left(\frac{\log \log^2 n}{\log^2 n}\right)$. There exists a randomized polynomial time algorithm that computes an order that is at least as likely as the original order with probability $1 - n^{-\alpha}$.*

Noisy comparisons. In the noisy comparisons model, Braverman and Mossel [6] compute the maximum likelihood permutation for $\gamma = \Omega(1)$. Here we improve their bound (without relaxing the solution concept) and compute the maximum likelihood for $\gamma = \tilde{\Omega}(\log^{-1/6} n)$.

Theorem 1.2. *For any constant $\alpha > 0$, given $\binom{n}{2}$ Boolean pairwise comparisons $q(i, j)$ with noise parameter $\gamma = \Omega\left(\frac{\log \log n}{\log n}\right)^{1/6}$, there is a deterministic polynomial-time algorithm that finds the maximum-likelihood order with probability $1 - n^{-\alpha}$ (over the sampling).*

1.3 Techniques.

Braverman and Mossel’s dynamic programming paradigm is the following. First, construct some initial guess that is “close” to the true order. Then, enumerate over the $\text{poly}(n)$ -many ways that one can partition $[n]$ into two subintervals of length $n/2$ without deviating much from the initial guess. For each partition recurse (in a dynamic programming fashion to avoid exponential blowup in the running time).

The main new ingredient in our analysis, which allows us to improve over [6, 7], is a different definition of *close* (Definition 2.4). In [6, 7], distance is defined as the *maximum offset* of any element. Our new definition was inspired by the discussion in [16], on combining arbitrary single vertex moves with local optimization. We relax the notion of maximal distance: we require that *all* elements are within $\tilde{O}(\log^2 n)$ of their position in the true permutation, but only require *most* elements to be within $O(\log n)$. In other words, we allow a small number of “bad” elements to have an offset greater than $O(\log n)$, without incurring a super-polynomial time complexity.

Mallow’s model. One important limitation of the algorithm of Braverman and Mossel [7] for Mallow’s model is that its noise tolerance does not grow asymptotically with the number of samples, r .⁴ In particular, their initial guess distribution is made by taking, for each element, its average location across the sample permutations. When considering a large number of samples, it becomes difficult to reason about the average location. For example, note that the average location is a biased estimator: the expected average location of the first element is not 1, even as the number of permutations goes to ∞ .

This asymmetry near the ends of the permutation also makes analysis significantly harder. In contrast, the distribution of average location for an element that is infinitely far from both ends is much nicer. In order to avoid such edge effects, a key step in our algorithm is *adding noise* to the samples: We simulate the nicer distribution by “mixing” each of our samples permutations with $2T$ dummy elements (where “inifitely far” is replaced by a sufficiently large $T = \text{poly}(r, \log n)$). We show that the posterior distribution generated by this process is identical to the permutation in Mallow’s model of a permutation ρ' , where ρ' consists of T dummy elements followed by the n real elements in their true (unknown state of nature) order, and finally T additional dummy elements. Taking the average location of each element with respect to the “resampled” permutations yields a well behaved (unbiased and concentrated) estimator that we can use to obtain provable guarantees on our initial guess.

Noisy comparisons For the noisy comparisons model, there is an additional challenge of aggregating local information (the comparisons) into a crude, initial global structure (a permutation that is close to the true order). We follow Braverman and Mossel [6] and construct the initial guess permutation inductively. In particular, we show that the optimal (maximum-likelihood) permutation over $k-1$ elements, together with the optimal placement of the k^{th} element will be, with high probability, close to the optimal permutation over k elements. At each step, in order to find the optimal permutation, we use the same dynamic program we previously described. Our new definition of “close” allows us to improve the noise tolerance; the analysis is more involved.

1.4 Related Work

Related models. The work of Chierichetti et al. [8] is closely related to this paper: they study the reconstruction problem in a generalization of Mallow’s model: each expert has a different noise parameter; furthermore, the noise parameters may be unknown. As previously mentioned, they focus on the low noise-to-samples regime: in particular, for uniform noise parameter β they that require $\min\{\beta^2, 1\} \cdot r = \Omega(\log n)$. It would be interesting to generalize our results to the setting of different noise parameters.

One variation of the noisy comparisons model allows for multiple independent queries of the same edge. Note that in this model one can recover the true unerlying permutation exactly, with high probability. It is easy to solve this problem in time $O\left(n \log n \cdot \log \frac{n}{\gamma}\right)$; Feige et al. [11] give an algorithm that solves the problem in time $O\left(n \log \frac{n}{\gamma}\right)$.

Proccacia et al. [21] consider a unified *noisy choice model* which includes both Mallow’s and the noisy comparison model. They discuss methods for finding the maximum-likelihood estimates to

⁴Specifically, their analysis assumes that r is constant [7, Claim 19].

different properties of the true order, such as finding the top k elements. It is interesting to note that in contrast with our results, their methods perform better when the noise is *larger*.

Worst case analysis. Without any assumptions on the generating processes, both problems are NP-hard. The problem that corresponds to the noisy comparisons model is *Feedback Arc Set* (FAS). Given some graph $G = (V, E)$, the FAS problem is to find a minimal set of edges whose removal (or reversal) causes G to be acyclic. This well-studied problem was one of Karp’s original 21 NP-complete problems [14]; Karp uses a gap-preserving reduction from Vertex Cover, so the 1.36... (NP) and 2 (Unique Games) hardness of the approximation results of [9, 17] hold. An $O(\log n \log \log n)$ -approximation was given by Even et al. [10]. Furthermore, algorithms with additive approximation guarantees were given by Arora et al. [3] and Frieze and Kannan [12]. Alon [2] proved that FAS is also NP-hard when the input is an unweighted tournament; a PTAS for this case was given by Mathieu and Schudy [16].

Bartholdi, Tovey and Trick [4] showed that reconstructing an optimal permutation is NP-hard when the input is an average of permutations. A number of heuristics have been suggested in the statistical literature, e.g., [18, 20], and a PTAS was shown by Mathieu and Schudy [16]. Ailon [1] showed how to build upon [16] to get a learning algorithm for the problem that requires access to fewer labels.

Fixed parameter algorithms. Fixed-parameter polynomial time algorithms that use a similar dynamic program to [6, 7] were given by Betzler et al. [5] and Simjour [22] for Mallow’s reconstruction problem. Both results require time complexity that is exponential in the average Kemeny distance from the optimal permutation (where the average is taken over the samples).

These results have been improved by Karpinski and Schudy [15] who achieve time complexity exponential in square-root of the average Kemeny distance; their algorithm also works for the tournament-FAS problem (which corresponds to the noisy comparisons model).

2 Notations and Parameters

We denote the set of integers $\{1, 2, \dots, n\}$ by $[n]$. We assume that there are n items, labeled $\{1, 2, \dots, n\}$, with some (true) permutation ρ . When we sample a permutation, we denote the samples by π_1, \dots, π_r , and the optimal permutation, relative to the measure we use, σ . Note that σ does not necessarily equal ρ . We denote by $\pi(a)$ the location of item a under permutation π , and $\pi^{-1}(k)$ to denote the k^{th} item under π .

2.1 Reconstruction Settings

Definition 2.1. *In Mallow’s original reconstruction problem, we are given n , β , and r permutations, π_1, \dots, π_r , generated independently at random from some permutation ρ according to Mallow’s model. The output is a maximum likelihood permutation σ that maximizes*

$$\prod_{i=1}^r \Pr[\pi_i | \sigma] = \frac{1}{(Z(\beta, n))^r} e^{-\beta \sum_{i=1}^r d_K(\pi_i, \sigma)}. \quad (3)$$

Notice that this is equivalent to finding the permutation σ that minimizes

$$d(\sigma) = \sum_{i=1}^r d_K(\pi_i, \sigma). \quad (4)$$

Definition 2.2. In the simple noisy signal aggregation (SNSA) problem, [7] (called noisy sorting without resampling in [6]), the input consists of binary queries $q(i, j) \in \{+, -\}$, one for each pair $i \neq j \in [n]$, such that $q(i, j) = -q(j, i)$. Given a permutation σ , the score of σ is

$$s_q(\sigma) = \sum_{i, j: \sigma(i) < \sigma(j)} q(i, j). \quad (5)$$

The output is a maximum-likelihood permutation: a permutation that maximizes (5).

Remark 2.3. Note that the maximum-likelihood permutation (for either problem) is not necessarily unique; nor is it necessarily the true order.

2.2 Proximity of Permutations

We use the following measure of “closeness”.

Definition 2.4. Let $\sigma, \pi: [n] \rightarrow [1, n]$ be real-valued functions such that

1. **Hard bound:** The position of every element is somewhat close in σ and π :

For any $k \in [n]$,

$$|\pi(k) - \sigma(k)| \leq \Lambda_1;$$

2. **Soft bound:** The position of almost every element is very close:

For every continuous interval $I \subset [n]$ of size Λ_1 , for all but d elements k such that $\pi(k) \in I$,

$$|\pi(k) - \sigma(k)| \leq \Lambda_2.$$

Then, we say that σ is $(\Lambda_1, \Lambda_2, d)$ -close to π .

Note that our definition of $(\Lambda_1, \Lambda_2, d)$ -close is asymmetric; but this asymmetry is only up to a constant factor. Notice also that for this definition to make sense, σ, π don’t have to be permutations.

2.3 Parameters

Let $\alpha > 0$ be a constant. Let $\delta = \Theta(\frac{\log n}{\log \log n})$. For the “hard bound”, we let $\Lambda_1 = \Theta(\delta \log n)$. For the “soft bound”, $\Lambda_2 = \Theta(\delta \log \log n)$, and $d = \max\{\frac{e\Lambda_1}{2 \log n}, \frac{\log n}{\log \log n} (\alpha + 1)\}$. The reason for these particular choices will be made clear in the proofs.

2.4 Probability Distributions

We denote a random variable X sampled from the geometric distribution with parameter p (the probability of success of each coin flip) by $X \sim \mathbf{Geom}(p)$. Recall that $\mathbb{E}[X] = \frac{1-p}{p}$. We denote a random variable Y sampled from the Binomial distribution with parameters n and p by $Y \sim \mathbf{B}(n, p)$. We denote a random variable Z sampled from the Negative Binomial distribution with parameters r and p by $Z \sim \mathbf{NB}(r, p)$.

We use the following well-known facts connecting the geometric, Negative Binomial and Binomial distributions (e.g. [13]).

Fact 2.5. *The sum of r i.i.d. $\mathbf{Geom}(p)$ random variables is a $\mathbf{NB}(r, p)$ random variable.*

Fact 2.6. *If $Z \sim \mathbf{NB}(r, p)$ and $W \sim \mathbf{B}(r + k, p)$, then $\Pr(Z > k) = \Pr(W < r)$.*

3 Sorting an Almost Sorted List

In this section we assume that we have a permutation π that is “not too far” from the optimal permutation, and show how to obtain an optimal permutation from it, using dynamic programming. In Section 4 we show how to obtain such a permutation π when we are given r permutations generated according to Mallow’s model, and in Section 5, we show how to obtain such a permutation when our input is in the form of a tournament generated according to the noisy comparison model. Our main result for this section is the following theorem.

Theorem 3.1. *Let $[n]$ be a set of elements, let q be some scoring function on pairs of elements, and let π be a permutation on $[n]$. Then there exists an algorithm that finds a permutation σ that maximizes*

$$s_q(\sigma) = \sum_{i,j:\sigma(i)<\sigma(j)} q(i,j),$$

among all permutations that are $(\Lambda_1, \Lambda_2, d)$ -close to π .

The algorithm runs in time $O(n \cdot \Lambda_1^{6d} \cdot 2^{6\Lambda_2} \cdot \Lambda_2^2)$. In particular, when Λ_1, Λ_2 and d are as in Subsection 2.3, the algorithm runs in polynomial time.

In Mallow’s model, the score function is $q(i, j) = \#\{t : \pi_t(i) < \pi_t(j)\}$, where π_1, \dots, π_r are the observed permutations. It is easy to verify that minimizing this is equivalent to minimizing Equation (2). In the noisy comparisons model, the score function is the the number of disagreeing comparisons (Equation (5)).

To connect Theorem 3.1 with later results, we remark that in Mallow’s model, π corresponds to the order induced by average of the samples (that we will refer to as $\text{ORDER}_{\frac{\pi}{r}}$); in the noisy comparison model, π corresponds to the permutation generated from the previous iteration of the inductive algorithm.

Proof of Theorem 3.1. Our algorithm uses dynamic programming, recovering growing intervals of an optimal order. For ease of notation, assume without loss of generality that π is the identity permutation.

Let $i < j \in [n]$, and let I be the set of elements $k \in [n]$ satisfying $\sigma(i) \leq \sigma(k) \leq \sigma(j)$. We enumerate over all combinations of elements that can comprise I .

Let $I_1^- = \{i + \Lambda_1, \dots, j - \Lambda_1\}$ and $I_1^+ = \{i - \Lambda_1, \dots, j + \Lambda_1\}$. By the hard bound (1), $I_1^- \subset I \subset I_1^+$. Let $I_2^- = \{i + \Lambda_2, \dots, j - \Lambda_2\}$ and $I_2^+ = \{i - \Lambda_2, \dots, j + \Lambda_2\}$. By the soft bound (2), $|I_2^+ \setminus I| \leq 2d$, and $|I \setminus I_2^-| \leq 2d$. Thus, to determine I we must take I_2^- , remove at most $2d$ elements from $I_2^- \setminus I_1^-$, for which we have Λ_1^{2d} choices; add some of the $4\Lambda_1$ elements from $I_2^+ \setminus I_2^-$, $2^{4\Lambda_2}$ more choices; and finally add at most $2d$ elements from $I_1^+ \setminus I_2^+$ - another factor of Λ_1^{2d} . This gives a total of $\Lambda_1^{4d} \cdot 2^{4\Lambda_2}$ possibilities of choosing I .

From here, we follow the dynamic programming algorithm of Braverman and Mossel [6], which we summarize here for completeness. Assume w.l.o.g. that n is a power of 2. Now, let $J_1 = [n]$, $J_2 = \{1, \dots, n/2\}$, $J_3 = \{n/2 + 1, \dots, n\}$, ... J_2 and J_3 are the left and right halves of J_1 . J_4 and J_5 will be the left and right halves of J_2 , and so on, up to J_{n-1} . (The smallest intervals have 2 elements each.) For each $J_t = \{i, \dots, j\}$, consider the set of $\Lambda_1^{4d} \cdot 2^{4\Lambda_2}$ possible intervals $J_t' = \{\sigma(i), \dots, \sigma(j)\}$. We use dynamic programming to store an optimal ordering σ' for each J_t' , with its score. Note that there are at most $n \cdot \Lambda_1^{4d} \cdot 2^{4\Lambda_2}$ sets J_t' .

We start from J_{n-1} and continue iteratively to J_1 . The optimal order for each of the smallest intervals can be found in time $O(1)$. For a larger interval J_t and candidate set J_t' , consider all $\Lambda_1^{2d} \cdot 2^{2\Lambda_2}$ possibilities for dividing J_t' to left and right halves, respectively J_l' and J_r' . Since we already have the optimal internal order for J_l' and J_r' , we need only to consider the edges crossing from J_l' to J_r' .

The running time is

$$\begin{aligned} & \sum_{i=1}^{\log n} (\# \text{ intervals of length } 2^i) \cdot (\# \text{ checks}) \cdot (\text{cost of each check}) \\ &= \sum_{i=1}^{\log n} O\left(\frac{n\Lambda_1^{4d}2^{4\Lambda_2}}{2^i} \cdot \Lambda_1^{2d}2^{2\Lambda_2} \cdot 2^{2i}\right) = O\left(n^2\Lambda_1^{6d}2^{6\Lambda_2}\right). \end{aligned}$$

In fact, the cost of each check can be reduced from 2^{2i} to $O(\Lambda_2^2)$ by ignoring all edges between pairs of indices that are more than $2\Lambda_2$ away.

The final running time is therefore $O(n \cdot \Lambda_1^{6d} \cdot 2^{6\Lambda_2} \cdot \Lambda_2^2)$. \square

4 Noisy Order Aggregation (Mallow's Model)

Theorem 4.1. *Let π_1, \dots, π_r be permutations on n elements, generated independently according to Mallow's model with noise parameter β . Let $\alpha > 0$ be a parameter, and set $\delta = \frac{\alpha+1}{\min\{r, \beta\sqrt{r}\}}$. Then for $\delta = O\left(\frac{\log n}{\log \log n}\right)$, there is a randomized algorithm that computes an order that is at least as likely as the original order with probability $1 - n^{-\alpha}$ (over the sampling process and the randomness of the algorithm) in time $\text{poly}(n, r)$.*

Proof outline.

Given the algorithm for sorting an almost sorted list (Theorem 3.1), the remaining task is to find a good initial almost sorted list. The obvious choice - simply taking the average location of each element in the permutations - works for lower noise levels [7], but when the number of samples r grows larger, it is hard to reason about the average location. In particular, the behavior near the

boundaries (elements whose true index is close to 1 or n) is unintuitive: the expected location of the first element is greater than 1...

Let ρ denote the true order over $\{1, \dots, n\}$, and consider r independent permutations of

$$(-\infty, \dots, 0, \rho, n+1, \dots, \infty),$$

generated according to Mallow’s model (with same noise parameter β). Note that we use this notation in this section for simplicity; more formally, we mean the limit distribution for permutations of $(-\tau+1, \dots, 0, \rho, n+1, \dots, \tau+n)$ as $\tau \rightarrow \infty$.⁵

Assume that we had access to the locations of elements $(1, \dots, n)$ in those permutations. This ideal distribution does not suffer from the “boundary effects” mentioned in the last paragraph, and we would be able to guarantee that the average location of each element is an unbiased and nicely concentrated estimator of its original location in ρ (Subsection 4.4). Of course, in order to take the average position as an estimator we need to generate the samples from the ideal distribution. This is difficult for two reasons. First, it is not clear how to simulate Mallow’s model noise over infinitely many elements. Instead, we sample r independent permutations over $(-T+1, \dots, 0, \rho, n+1, \dots, T+n)$; in Subsection 4.3 we show that when taking $T = \text{poly}(r, \log n)$, the distribution over locations of elements $\{1, \dots, n\}$ is close (in total variation distance) to the ideal distribution over $(-\infty, \dots, \infty)$. The second problem is that we don’t have the original order ρ . Instead, for each $i \in [r]$ we sample a permutation over $(-T+1, \dots, 1, \dots, n, \dots, T+n)$, and then replace elements $\{1, \dots, n\}$ with the input sample π_i . In Subsection 4.2 we argue that when the π_i ’s are generated according to Mallow’s model, this “re-sampling” procedure results in the same distribution over random permutations of $(-T+1, \dots, 0, \rho, n+1, \dots, T+n)$.

4.1 The Standard Generating Process

In this subsection we recall the standard iterative generating process of Mallow’s model (e.g. [7]), and define some notation. For simplicity and w.l.o.g., we assume that the true order is $\rho = (1, \dots, n)$. A standard way of generating samples according to Mallow’s model is by induction. Suppose that we sampled a permutation π^{k-1} over $\{1, \dots, k-1\}$. We sample a permutation π^k over $\{1, \dots, k\}$ by inserting k into π^{k-1} at a random location. By (1), we have that k is most likely to be placed last in π^k ; it is $e^{-\beta}$ as likely to be placed second-to-last (in the $(k-1)$ -th location); $e^{-2\beta}$ as likely to be placed in location $k-2$, and so on. In other words, we place element k in location $k - \eta(k)$, where $\eta(k)$ is a random *displacement* drawn according to a *truncated* Geometric distribution with parameter $1 - e^{-\beta}$, with the truncation at $k-1$ because an element cannot have a negative location. (In other words, $\eta(k)$ is sampled from $\mathbf{Geom}(1 - e^{-\beta})$ conditioned on $\eta(k) \leq k-1$.) The purpose of our re-sampling is precisely to nullify this truncation.

Notice that once we place element k in location $k - \eta(k)$, it can still shift to a larger-index location when it gets “bumped” by future insertions. We denote the total displacements in that direction $\eta^{\text{rev}}(k)$; notice that $\eta^{\text{rev}}(k)$ is also a truncated Geometric random variable (truncated at $n - k - 1$).

The final location of element k is:

$$\pi(k) = k - \eta(k) + \eta^{\text{rev}}(k).$$

⁵The discussion in Subsection 4.3 asserts that this limit exists.

4.2 The Re-Sampling Generating Process

We would like to generate Mallow's model permutations from $(-T + 1, \dots, 0, \rho, n + 1, \dots, T + n)$. One method is the standard generating process described in the previous subsection (by induction, starting from $-T + 1$, then $\{-T + 1, -T + 2\}$, etc). Denote the resulting distribution of this process by D_1 .

We describe another generating process for the same distribution:

1. Sample a permutation π^T of $(-T + 1, \dots, 0, 1, \dots, n, n + 1, \dots, T + n)$ using the generating process from the Subsection 4.1.
2. Independently, sample a permutation π of ρ using the same procedure.
3. Rearrange the occurrences of $\{1, \dots, n\}$ in π^T according to π (keeping the locations of all other elements in $\{-T + 1, \dots, T + n\} \setminus \{1, \dots, n\}$); denote this final permutation by π' .

To illustrate this resampling procedure, consider the following example. Let $T = 2$ and $n = 3$, and suppose that we sampled the permutations $\pi^T = (3, -1, 5, 0, 1, 2, 4)$ and $\pi = (2, 1, 3)$. We ignore the order in which $\{1, 2, 3\}$ arrived in π^T , remembering only their locations: $(*, -1, 5, 0, *, *, 4)$, and insert them back according to π ; that is, $\pi' = (2, -1, 5, 0, 1, 3, 4)$. Notice that, given access to π , we did not need to know the original ρ .

Let D_2 denote the distribution induced by the latter generating process. We now argue that $D_1 = D_2$. We need the following observation.

Observation 4.2. *Let i and j be two elements that are adjacent in ρ . Let σ and σ' be two permutations that are identical except that the locations of i and j are exchanged, and σ agrees with ρ on the relative order of $\{i, j\}$. Then*

$$\Pr[\sigma|\rho] = e^{-\beta} \Pr[\sigma'|\rho].$$

Proof. It is easy to see that $d_K(\sigma', \rho) = d_K(\sigma, \rho) + 1$, as the exchange in position adds one to the Kemeny distance (see Equation (2)). The result follows immediately from Equation (1). □

Claim 4.3. $D_1 = D_2$.

Proof. We partition the permutations over $\{-T + 1, \dots, T + n\}$ into $(2T + n)!/n!$ equivalence classes where $[\pi^T]$ contains all the permutations that agree with π^T on the location of all elements not in $\{1, \dots, n\}$. We first observe that D_1 and D_2 assign the same total weight to each equivalence class since the two generating processes act identically on elements not in $\{1, \dots, n\}$.

Next, we argue that the relative weight that D_1 and D_2 assign to each permutation in $[\pi^T]$ is the same (by the previous paragraph, this proves the claim). Furthermore, it suffices to prove for pairs of “adjacent” permutations as in Observation 4.2, as we can move from any permutation $\sigma_1 \in [\pi^T]$ to any permutation $\sigma_2 \in [\pi^T]$ using a finite number of such operations. Finally, for any such pair of permutations, we apply Observation 4.2 to both generating processes, so the ratio of probabilities assigned by both distributions is e^β . □

4.3 Bounding T

In this section we argue that when $\beta^2 r = \Omega\left(\frac{\log \log^2 n}{\log^2 n}\right)$, it suffices to take $T = \text{poly}(r, \log n)$ in order to simulate the distribution of locations of $\{1, \dots, n\}$ in permutations of $(-\infty, \dots, 0, \rho, n+1, \dots, \infty)$ with error probability $\text{poly}(1/n)$.

Claim 4.4. *For any $\tau \in \mathbb{N}$, let D_τ denote the joint distribution of locations of $\{1, \dots, n\}$ when permuting $(-\tau+1, \dots, 0, \rho, n+1, \dots, \tau+n)$ according to Mallow's model with parameter β . Then for any $\tau_1, \tau_2 \geq T$, D_{τ_1} and D_{τ_2} are within $2n \cdot e^{-\beta T}$ of each other in total variation distance.*

Notice, in particular, that for $T > \ln(2n^{\alpha+1}r)/\beta$, this implies total variation distance of at most $n^{-\alpha}/r$; by union bound this implies that the joint distributions over r independent samples are within $n^{-\alpha}$.

Proof sketch. It suffices to argue that for every $k \in \{1, \dots, n\}$, the offsets $\eta(k)$ and $\eta^{\text{rev}}(k)$ are distributed approximately the same in both cases. Since they are both truncated Geometric variables, this reduces to showing that they reach the truncation threshold (which is at least T) with very low probability:

$$\Pr_{X \sim \text{Geom}(1-e^{-\beta})} [X > T] < e^{-\beta T}.$$

The claim follows by taking a union bound over all $2n$ displacements. \square

4.4 Analyzing the Ideal Distribution

From now on, we focus our attention on the locations of $\{1, \dots, n\}$ in random permutations of $(-\infty, \dots, 0, \rho, n+1, \dots, \infty)$. The first nice property of this distribution is that by symmetry $\pi'(k)$ is an unbiased estimator of $\rho(k)$, i.e.,

$$\mathbb{E}[\pi'(k)] = \rho(k). \quad (6)$$

More importantly, for r samples π'_1, \dots, π'_r , we show that their average $\overline{\pi'} = \frac{1}{r} \sum \pi'_t(k)$ concentrates around its expectation $\rho(k)$. We do this by arguing that the average offset $\overline{\eta'} = \frac{1}{r} \sum \eta'_t(k)$ concentrates around its expectation. Since $\eta'_t(k)$ is drawn from Geometric distribution with parameter $1 - e^{-\beta}$, its expectation is

$$\mathbb{E}[\eta'_t(k)] = \frac{e^{-\beta}}{1 - e^{-\beta}}.$$

Lemma 4.5. *For any $i, k \in [n]$*

$$\Pr \left[\left| \overline{\eta'}(k) - \mathbb{E}[\overline{\eta'}(k)] \right| \geq i \right] < 2 \exp \left(-\frac{r}{162} \cdot \min \left\{ (i\beta)^2, i\beta, i \right\} \right).$$

Proof. Sums of i.i.d. geometric variables (such as $\overline{\eta'}(k)$) are convenient because they behave like Negative Binomial random variables (Fact 2.5). This allows us to use the Chernoff bound to bound the deviation from expectation of each sum of geometric variables.

Our proof splits into two cases: when $\beta \leq 1/2$, and when $\beta > 1/2$. For each of these, we bound separately the two possibilities defined by the absolute value.

Case $\beta \leq 1/2$, positive

We first transform our event from a question about a sum of geometric variables to a question about a binomial variable. Using Facts 2.5 and 2.6,

$$\begin{aligned}
\Pr \left[r\bar{\eta}'(k) \geq r\mathbb{E}[\bar{\eta}'(k)] + ri \right] &= \Pr \left[\mathbf{NB}(r, (1 - e^{-\beta})) \geq \frac{re^{-\beta}}{(1 - e^{-\beta})} + ri \right] \\
&= \Pr \left[\mathbf{NB}(r, (1 - e^{-\beta})) \geq \left\lceil \frac{re^{-\beta}}{(1 - e^{-\beta})} + ri \right\rceil \right] \\
&= \Pr \left[\mathbf{B} \left(\left\lceil \frac{r}{1 - e^{-\beta}} + ri \right\rceil - 1, 1 - e^{-\beta} \right) < r \right]. \tag{7}
\end{aligned}$$

The expectation μ of this Binomial distribution satisfies

$$r + ri \cdot (1 - e^{-\beta}) - 1 \leq \mu \leq 2r + ri \cdot (1 - e^{-\beta}).$$

Therefore, for its value to be less than r , it must deviate from the mean by at least

$$\Delta = ri \cdot (1 - e^{-\beta}). \tag{8}$$

We would like to use the Chernoff bound, for which we compute the value of Δ^2/μ :

$$\begin{aligned}
\frac{\Delta^2}{\mu} &\geq \frac{(ri)^2 (1 - e^{-\beta})^2}{2r + ri \cdot (1 - e^{-\beta})} \\
&\geq \frac{r}{4} \cdot \min \left\{ i^2 (1 - e^{-\beta})^2, i (1 - e^{-\beta}) \right\}. \tag{9}
\end{aligned}$$

We now use the fact that $\beta \leq 1/2$ to bound $(1 - e^{-\beta})$: Applying the mean value theorem to $(1 - e^{-\beta})$, we have that for some $\beta' \in [0, \beta]$, $(1 - e^{-\beta}) = \beta \cdot e^{-\beta'}$. By our assumption that $\beta \leq 1/2$, we have $e^{-\beta'} \geq e^{-1/2} > 1/2$, and hence

$$1 - e^{-\beta} > \beta/2. \tag{10}$$

Plugging into (8), this implies $\Delta > ri\beta/2$. Also, plugging (10) into (9), we get:

$$\begin{aligned}
\frac{\Delta^2}{\mu} &> \frac{r}{4} \cdot \min \left\{ i^2 \left(\frac{\beta}{2} \right)^2, i \left(\frac{\beta}{2} \right) \right\} \\
&\geq \frac{r}{16} \cdot \min \left\{ (i\beta)^2, i\beta \right\}.
\end{aligned}$$

Therefore, by the Chernoff bound

$$\begin{aligned}
\Pr \left[\bar{\eta}'(k) \geq r\mathbb{E}[\bar{\eta}'(k)] + ri \right] &\leq \exp \left(- \min \left\{ \frac{\Delta}{3}, \frac{\Delta^2}{3\mu} \right\} \right) \\
&\leq \exp \left(- \frac{r}{48} \cdot \min \left\{ (i\beta)^2, i\beta \right\} \right).
\end{aligned}$$

Case $\beta \leq 1/2$, negative

Similarly, using Facts 2.5 and 2.6,

$$\begin{aligned}
\Pr \left[r\bar{\eta}'(k) \leq r\mathbb{E}[\bar{\eta}'(k)] - ri \right] &= \Pr \left[\mathbf{NB}(r, (1 - e^{-\beta})) \leq \frac{re^{-\beta}}{(1 - e^{-\beta})} - ri \right] \\
&= \Pr \left[\mathbf{NB}(r, (1 - e^{-\beta})) \leq \left\lfloor \frac{re^{-\beta}}{(1 - e^{-\beta})} - ri \right\rfloor \right] \\
&= \Pr \left[\mathbf{B} \left(\left\lfloor \frac{r}{1 - e^{-\beta}} - ri \right\rfloor, 1 - e^{-\beta} \right) \geq r \right]. \tag{11}
\end{aligned}$$

The expectation μ of this Binomial distribution is at most

$$\mu \leq r - ri \cdot (1 - e^{-\beta}).$$

Therefore, for its value to be at least r , it must deviate from the mean by at least

$$\Delta = ri \cdot (1 - e^{-\beta}) \geq ri\beta/2.$$

We would like to use the Chernoff bound, for which we compute the value of Δ^2/μ :

$$\begin{aligned}
\frac{\Delta^2}{\mu} &\geq \frac{(ri)^2 (1 - e^{-\beta})^2}{r + ri \cdot (1 - e^{-\beta})} \\
&\geq \frac{r}{2} \cdot \min \left\{ i^2 (1 - e^{-\beta})^2, i (1 - e^{-\beta}) \right\} \\
&\geq \frac{r}{4} \cdot \min \left\{ i^2 \left(\frac{\beta}{2} \right)^2, i \left(\frac{\beta}{2} \right) \right\} \tag{Ineq. (10)} \\
&\geq \frac{r}{8} \cdot \min \left\{ (i\beta)^2, i\beta \right\}.
\end{aligned}$$

Therefore, by the Chernoff bound

$$\begin{aligned}
\Pr \left[\bar{\eta}'(k) \leq r\mathbb{E}[\bar{\eta}'(k)] - ri \right] &\leq \exp \left(- \min \left\{ \frac{\Delta}{3}, \frac{\Delta^2}{3\mu} \right\} \right) \\
&\leq \exp \left(- \frac{r}{24} \cdot \min \left\{ (i\beta)^2, i\beta \right\} \right).
\end{aligned}$$

Case $\beta > 1/2$, positive

Continuing from Inequality (7), we consider the complement Binomial distribution:

$$\begin{aligned}
\Pr \left[r\bar{\eta}'(k) \geq r\mathbb{E}[\bar{\eta}'(k)] + ri \right] &= \Pr \left[\mathbf{B} \left(\left\lfloor \frac{r}{1 - e^{-\beta}} + ri \right\rfloor - 1, 1 - e^{-\beta} \right) < r \right] \\
&= \Pr \left[\mathbf{B} \left(\left\lfloor \frac{r}{1 - e^{-\beta}} + ri \right\rfloor - 1, e^{-\beta} \right) \geq \left\lfloor \frac{r}{1 - e^{-\beta}} + ri \right\rfloor - r \right] \\
&= \Pr \left[\mathbf{B} \left(\left\lfloor \frac{r}{1 - e^{-\beta}} + ri \right\rfloor - 1, e^{-\beta} \right) \geq \left\lfloor \frac{re^{-\beta}}{1 - e^{-\beta}} + ri \right\rfloor \right].
\end{aligned}$$

The deviation Δ remains the same as in (8):

$$\Delta \geq ri \cdot (1 - e^{-\beta}), \quad (12)$$

and the expectation now satisfies:

$$\mu \leq \frac{re^{-\beta}}{1 - e^{-\beta}} + ri \cdot e^{-\beta}.$$

We now have,

$$\begin{aligned} \Delta^2/\mu &= \frac{(ri)^2 (1 - e^{-\beta})^2}{\frac{re^{-\beta}}{1 - e^{-\beta}} + ri \cdot e^{-\beta}} = \frac{i^2 (1 - e^{-\beta})^2}{\frac{1}{1 - e^{-\beta}} + i} \cdot re^{\beta} \\ &\geq \frac{re^{\beta}}{2} \cdot \min \left\{ i^2 (1 - e^{-\beta})^3, i (1 - e^{-\beta})^2 \right\}. \end{aligned}$$

For $\beta > 1/2$, we have $1 - e^{-\beta} > 1 - e^{-1/2} > 1/3$. Furthermore, for any $\beta > 0$, $e^{\beta} > \beta, \beta^2$. Thus, $\Delta \geq ri/3$, and also:

$$\begin{aligned} \Delta^2/\mu &> \frac{r}{2} \cdot \min \left\{ (i\beta)^2 (1/3)^3, (i\beta) (1/3)^2 \right\} \\ &> \frac{r}{54} \cdot \min \left\{ (i\beta)^2, i\beta \right\}. \end{aligned}$$

Finally, applying the Chernoff bound we have

$$\begin{aligned} \Pr \left[r\bar{\eta}'(k) \geq r\mathbb{E}[\bar{\eta}'(k)] + ri \right] &\leq \exp \left(- \min \left\{ \frac{\Delta}{3}, \frac{\Delta^2}{3\mu} \right\} \right) \\ &\leq \exp \left(- \frac{r}{162} \cdot \min \left\{ (i\beta)^2, i\beta, i \right\} \right). \end{aligned}$$

Case $\beta > 1/2$, negative

Similarly, starting from (11) and looking at the complement Binomial distribution, we have:

$$\begin{aligned} \Pr \left[r\bar{\eta}'(k) \leq r\mathbb{E}[\bar{\eta}'(k)] - ri \right] &= \Pr \left[\mathbf{B} \left(\left\lfloor \frac{r}{1 - e^{-\beta}} - ri \right\rfloor, 1 - e^{-\beta} \right) \geq r \right] \\ &= \Pr \left[\mathbf{B} \left(\left\lfloor \frac{r}{1 - e^{-\beta}} + ri \right\rfloor, e^{-\beta} \right) \leq \left\lfloor \frac{r}{1 - e^{-\beta}} + ri \right\rfloor - r \right] \\ &= \Pr \left[\mathbf{B} \left(\left\lfloor \frac{r}{1 - e^{-\beta}} + ri \right\rfloor, e^{-\beta} \right) \leq \left\lfloor \frac{re^{-\beta}}{1 - e^{-\beta}} + ri \right\rfloor \right]. \end{aligned}$$

The deviation Δ remains:

$$\Delta \geq ri \cdot (1 - e^{-\beta}) \geq ri/3, \quad (13)$$

and the expectation satisfies:

$$\mu \leq \frac{re^{-\beta}}{1 - e^{-\beta}} + ri \cdot e^{-\beta}.$$

We now have,

$$\begin{aligned} \Delta^2/\mu &= \frac{(ri)^2 (1 - e^{-\beta})^2}{\frac{2re^{-\beta}}{1 - e^{-\beta}} + ri \cdot e^{-\beta}} = \frac{i^2 (1 - e^{-\beta})^2}{\frac{1}{1 - e^{-\beta}} + i} \cdot re^{\beta} \\ &\geq \frac{re^{\beta}}{2} \cdot \min \left\{ i^2 (1 - e^{-\beta})^3, i (1 - e^{-\beta})^2 \right\} \\ &\geq \frac{r}{2} \cdot \min \left\{ (i\beta)^2 (1/3)^3, (i\beta) (1/3)^2 \right\} && \text{(Using } \beta > 1/2) \\ &> \frac{r}{54} \cdot \min \left\{ (i\beta)^2, i\beta \right\}. \end{aligned}$$

Finally, applying the Chernoff bound we have

$$\begin{aligned} \Pr \left[r\bar{\eta}^{\prime}(k) \geq r\mathbb{E}[\bar{\eta}^{\prime}(k)] + ri \right] &\leq \exp \left(- \min \left\{ \frac{\Delta}{3}, \frac{\Delta^2}{3\mu} \right\} \right) \\ &\leq \exp \left(- \frac{r}{162} \cdot \min \left\{ (i\beta)^2, i\beta, i \right\} \right). \end{aligned}$$

□

4.5 Average Location in Ideal Distribution is Close to True Order

Suppose $\beta^2 r = \Omega \left(\frac{\log \log^2 n}{\log^2 n} \right)$. We let

$$\begin{aligned} L_1 &= \frac{1000(\alpha + 1)(\log n)}{\min\{r, \beta\sqrt{r}\}} = O \left(\frac{\log^2 n}{\log \log n} \right) \\ L_2 &= \frac{2000(\alpha + 1)(\log \log n)}{\min\{r, \beta\sqrt{r}\}} = O(\log n) \\ d &= \max \left\{ \frac{eL_1}{\log n}, \frac{\log n}{\log \log n} (\alpha + 1) \right\} = O \left(\frac{\log n}{\log \log n} \right) \end{aligned}$$

Lemma 4.6. *Let $\pi'_1, \dots, \pi'_r : [n] \rightarrow \mathbb{Z}$ denote the locations of $\{1, \dots, n\}$ in r independent permutations of $(-\infty, \dots, 0, \rho, n + 1, \dots, \infty)$ generated independently according to Mallow's model with noise parameter β , such that $\beta^2 r = \Omega \left(\frac{\log \log^2 n}{\log^2 n} \right)$. Then the average location $\bar{\pi}' = \frac{1}{r} \sum \pi'_t$ is, with probability $1 - 3n^{-\alpha}$, (L_1, L_2, d) -close to ρ .*

We first prove the hard bound for every element:

Claim 4.7 (Hard bound).

$$\Pr \left[\exists k \in [n] : \left| \bar{\pi}'(k) - \rho(k) \right| > L_1 \right] < n^{-\alpha}.$$

Proof. For each $k \in [n]$,

$$\begin{aligned} \Pr \left[\left| \overline{\pi'}(k) - \rho(k) \right| > L_1 \right] &= \Pr \left[\left| \overline{\pi'}(k) - \mathbb{E}[\overline{\pi'}(k)] \right| > L_1 \right] && \text{(Eq. (6))} \\ &< 4 \exp \left(-\frac{r}{162} \cdot \min \left\{ (L_1\beta)^2, L_1\beta, L_1 \right\} \right) && \text{(Lemma 4.5)} \\ &< n^{-\alpha-1} && \text{(Choice of } L_1\text{).} \end{aligned}$$

The claim follows by taking a union bound over all $k \in [n]$. \square

In order to prove the soft bound, we require one more definition.

Definition 4.8. We say that an index k is 1-to- n bad if the deviation of its average offset from expectation exceeds $O(\log n)$:

$$|\overline{\eta}(k) - E[\overline{\eta}(k)]| > L_2/2.$$

Similarly, k is n -to-1 bad if its average reverse offset is far from its expectation:

$$|\overline{\eta}^{\text{rev}}(k) - E[\overline{\eta}^{\text{rev}}(k)]| > L_2/2.$$

If k is 1-to- n bad or n -to-1 bad, then it is bad, and otherwise we say that it is good.

In particular, notice that if k is good, then $\left| \overline{\pi'}(k) - \rho(k) \right| \leq L_2$. Therefore, in order to prove the soft bound, it suffices to show that there are not too many bad elements.

Claim 4.9 (Soft bound). *With probability $1 - 2n^{-\alpha}$, for every interval $I \subset [n]$ of length L_1 which is contiguous in ρ , the number of bad elements is at most d .*

Proof. The events that different elements are good are not independent. However, the average offsets at insertion, $\overline{\eta}(k)$'s, are independent. In particular the event that any element is 1-to- n bad is independent from the the corresponding event for another element.

By symmetry,

$$\Pr [|\{\text{bad elements in } I\}| > d] \leq 2 \Pr [|\{1\text{-to-}n \text{ bad elements in } I\}| > d/2].$$

The probability that each element is 1-to- n bad is at most:

$$\begin{aligned} \Pr [|\overline{\eta}(k) - E[\overline{\eta}(k)]| > L_2/2] &< 4 \exp \left(-\frac{r}{162} \cdot \min \left\{ \left(\frac{L_2}{2} \beta \right)^2, \frac{L_2}{2} \beta, \frac{L_2}{2} \right\} \right) && \text{(Lemma 4.5)} \\ &< \log^{-4} n && \text{(Choice of } L_2\text{).} \end{aligned}$$

Using the independence of the 1-to- n bad events, we have that for each subset of $d/2$ elements, the probability that they are all 1-to- n bad is bounded by $\log^{-2d} n$.

For each I , take a union bound over all possible choices of d elements from I , to get:

$$\begin{aligned} \Pr [|\{1\text{-to-}n \text{ bad elements in } I\}| > d/2] &< \binom{L_1}{d/2} \cdot \log^{-2d} n \\ &\leq \left(\frac{eL_1}{d \log^2 n} \right)^d && \text{(Stirling's formula)} \\ &\leq (\log n)^{-\frac{\log n}{\log \log n}(\alpha+1)} = n^{-\alpha-1} && \text{(Choice of } d\text{).} \end{aligned}$$

\square

4.6 Obtaining a Candidate Permutation from the Average Location

We cannot directly apply the algorithm of Section 3 to our simple estimator $\overline{\pi'}$, because $\overline{\pi'}$ is not a permutation. However, as we argue in this subsection, we can simply order the elements by $\overline{\pi'}$ (breaking ties arbitrarily); we denote the resulting permutation by $\text{ORDER}_{\overline{\pi'}}$. As $\text{ORDER}_{\overline{\pi'}}$ is a permutation, we can feed it as an initial candidate to the sorting algorithm.

The following lemma shows that if $\overline{\pi'}$ is close to the true order (as guaranteed by Lemma 4.6), then so is $\text{ORDER}_{\overline{\pi'}}$.

Lemma 4.10. *Let L_1, L_2 and d be as in the previous subsection. With probability $1 - 3n^{-\alpha}$, $\text{ORDER}_{\overline{\pi'}}$ is $(2L_1, 2L_2 + 2d, 5d)$ -close to the true order ρ .*

Proof. We prove the hard and soft bounds separately.

Hard bound. By Lemma 4.6, if $\overline{\pi'}(b) \leq \overline{\pi'}(a)$, then

$$\rho(b) \leq \overline{\pi'}(b) + L_1 \leq \overline{\pi'}(a) + L_1 \leq \rho(a) + 2L_1. \quad (14)$$

Therefore there are at most $\rho(a) + 2L_1$ elements that appear before a in $\overline{\pi'}$; i.e. $\text{ORDER}_{\overline{\pi'}}(a) \leq \rho(a) + 2L_1$. The other direction follows by a symmetric argument.

Soft bound. For the soft bound, notice that if $\overline{\pi'}(b) \leq \overline{\pi'}(a)$, then either a or b are bad elements (as in Definition 4.8), or

$$\rho(b) \leq \overline{\pi'}(b) + L_2 \leq \overline{\pi'}(a) + L_2 \leq \rho(a) + 2L_2.$$

Therefore, there are at most $\rho(a) + 2L_2$ good elements that appear before a in $\text{ORDER}_{\overline{\pi'}}$. We also need to take into account the bad elements: by (14), a can only switch positions with elements whose original position is within $2L_1$ of $\rho(a)$. By the soft bound in Lemma 4.6, among the $2L_1$ elements z s.t. $\rho(a) < \rho(z) \leq \rho(a) + 2L_1$, there are at most $2d$ bad elements. Therefore if a is good, we have that

$$\left| \text{ORDER}_{\overline{\pi'}}(a) - \rho(a) \right| \leq 2L_2 + 2d.$$

Finally, by (14), the original positions of $\{a : \text{ORDER}_{\overline{\pi'}}(a) \in I\}$ belong to a contiguous subset in ρ of size at most $5L_1$; therefore at most $5d$ of those elements are bad elements. \square

5 Noisy Comparison Aggregation

In this section we prove our result for aggregation of noisy pairwise comparisons into an order.

Theorem 1.2. *For any constant $\alpha > 0$, given $\binom{n}{2}$ Boolean pairwise comparisons $q(i, j)$ with noise parameter $\gamma = \Omega\left(\frac{\log \log n}{\log n}\right)^{1/6}$, there is a deterministic polynomial-time algorithm that finds the maximum-likelihood order with probability $1 - n^{-\alpha}$ (over the sampling).*

The rest of this section is devoted to proving Theorem 1.2. For the noisy comparison model, we do not have a natural starting point like the average order $\overline{\pi}$ which we used for the noisy order aggregation. Instead, we reconstruct an optimal permutation by successively inserting elements into the “correct” position relative to the elements that we already placed. The proof of correctness is by induction: at step k , we assume the permutation over the first $k - 1$ elements is close to the true

order, and show that inserting the k -th element in the correct position guarantees that the permutation over the k elements is also close to the true order on those elements. This iterative construction is described in Section 5.2. Before we describe it, though, we show, in Section 5.1, that for any optimal order σ , the discrepancy between σ and the true order ρ , is bounded, where the distance between permutations is measured by

$$d(\pi_1, \pi_2) = \sum_{i=1}^n |\pi_1(i) - \pi_2(i)|.$$

5.1 Bounding the Discrepancy between the True and Optimal Orders

We first show that any optimal order is close, with high probability, to the true order. Similarly to before, in order to be able to apply the algorithm of Section 3, we set the parameters as follows: Let $\epsilon = n^{-(\alpha+2)}$, $L_1 = \Theta\left(\frac{\log^2 n}{\log \log n}\right)$ (we give a more precise formulation of L_1 in Equation (16)), $L_2 = \Theta(\log n)$, and $d = \max\left\{\frac{\epsilon L_1}{2 \log n}, -\log \epsilon\right\}$.

Proposition 5.1. *For ϵ as above, let σ be an optimal order relative to a tournament T generated according to the noisy comparisons model with parameter $\gamma = \Omega\left(\frac{\log \log n}{\log n}\right)^{1/6}$ from a permutation ρ . Then, with probability $1 - (2n + 1)\epsilon$, ρ is (L_1, L_2, d) -close to σ .*

The following lemma is used to bound the total distance of the optimal permutation from the true permutation. We notice that the lemma holds for sufficiently large subsets of $[n]$, in particular,

Lemma 5.2. (cf. Lemma 3.3 of [6])

Let $c \geq 1$ (possibly super-constant) and $I \subset [n]$, where n and I are sufficiently large. Suppose that

$$\gamma^2 c = \Omega(\ln c). \quad (15)$$

Then the probability that there exists an optimal permutation σ over I at distance $d(\sigma, \rho) > c_2 |I|$ is at most $2^{-c|I|\gamma^2/10}$.

Proof. (Sketch) The probability that any specific permutation beats the true order decays exponentially in its distance from the tournament. Taking a union bound over all permutations at each distance greater than $c|I|$ yields the bound. \square

In [6], γ is a constant, so we can also take c to be a constant. For our setting, we want set $c = \Theta(\log^{2/6} n)$ to support $\gamma = \Omega\left(\frac{\log \log n}{\log n}\right)^{1/6}$.

Braverman and Mossel use Lemma 5.2 to bound the offset of any single element by $O(\log n)$, using Lemma 5.3. This lemma is sufficient to obtain our hard bound; however we will need to be more careful in order to prove our soft bound.

Lemma 5.3. (Lemma 3.5 of [6])

With probability $1 - \epsilon$, for every optimal ordering σ and for every k ,

$$|\rho(k) - \sigma(k)| < L_1 = O(\gamma^{-4} \log n - \log \epsilon). \quad (16)$$

Lemma 5.3 provides us with the hard bound: $L_1 = O\left(\frac{\log^2 n}{\log \log n}\right)$ if $\gamma = \Omega\left(\frac{\log \log n}{\log n}\right)^{1/6}$. We use Lemma 5.2 in a more subtle way to prove the soft bound, which is a corollary to the following lemma.

Lemma 5.4. *Let σ be an optimal permutation. With probability $1 - 2\epsilon$, for every subset $H \subset [n]$ of size L_1 that is continuous in σ , the total distance over elements in H between σ and the true permutation ρ is bounded:*

$$\Pr \left[\sum_{h \in H} |\sigma(h) - \rho(h)| > L_2 \right] < 2\epsilon. \quad (17)$$

Proof. We would like to treat each subset H as a separate permutation, and apply Lemma 5.2 to bound the total dislocation distance between the true and optimal permutations on H . However, there are a few subtleties arising from the dependencies of the optimal permutation of H (and even merely the fact that H is continuous in some optimal σ), and elements outside H . We overcome those subtleties by considering paddings of H that isolate it from events related to other elements.

Consider a subset I of size $5L_1$ that is continuous in σ . The optimal ordering of vertices within I depends only on the comparisons between them. This allows us to analyze the internal structure of σ on I independently of the rest of the graph.

The choice of I as a continuous interval in the optimal permutation depends on the edges within I . Therefore, we consider a larger interval J , of size $7L_1$, and bound the average offset simultaneously for every candidate $I' \subset J$, whenever (16) holds.

Furthermore, notice that while the order between the elements of I in σ is close to their relative order in the true permutation, it does not guarantee that their absolute positions are also close: When the elements of I swap positions with elements not in I , they do not affect the total offset of the local permutation. Finally, consider the set H which is composed of the middle L_1 elements in I (according to σ). The elements in H can only swap positions with other elements in I , and therefore the total distance of the locally optimal permutation on I upper bounds the total distance between σ and the true permutation on H .

Suppose that $H = \{\sigma^{-1}(i), \dots, \sigma^{-1}(i + L_1)\}$, and let $I = \{\sigma^{-1}(i - 2L_1), \dots, \sigma^{-1}(i + 3L_1)\}$ be a padding of H of size $5L_1$. Whenever (16) holds, the true location of the elements in I satisfies $\rho(I) \subset \{i - 3L_1, \dots, i + 4L_1\} = J$.

Consider any subset $I' \subset J$ of size $|I'| = 5L_1$, and let $\sigma(I')$ denote optimal permutation on I' . Let c be such that Equation 15 holds (we will choose c more carefully in Corollary 5.5). By Lemma 5.2, the probability that the average distance between $\sigma(I')$ and the true permutation restricted to I' is greater than c is bounded:

$$\begin{aligned} \Pr [d(\sigma(I'), \rho(I')) > c \cdot 5L_1] &< 2^{-5cL_1\gamma^2/10} \\ &\leq 2^{-7L_1 + \log \epsilon} \leq \left(\frac{7L_1}{5L_1}\right)^{-1} \epsilon, \end{aligned}$$

for an appropriate selection of c . Taking a union bound over all $I' \subset J$ gives that

$$\Pr [d(\sigma(I'), \rho(I')) > c \cdot 5L_1] < \epsilon.$$

In particular, for I which is continuous in σ , the restriction of σ to I , $\sigma|_I$, is also an optimal order. We get that $\sigma|_I$ is close to the true order on I .

Now observe that if (16) holds, then if σ disagrees with the true order over some pair (h, i) , for $h \in H$, it must be that $i \in I$. Therefore, the offset of h under σ is exactly the same as its offset by

$\sigma|_I$ relative to the true offset by I . Therefore we get that except with probability $< 2\epsilon$,

$$\sum_{h \in H} |\sigma(h) - h| \leq d(\sigma|_I) \leq c \cdot 5L_1.$$

□

We can now use Markov’s inequality to bound the number of “bad” elements within each continuous interval.

Corollary 5.5. *Let σ be an optimal permutation. Except with probability 2ϵ (in particular, whenever the event in (17) holds), for every subset $H \subset [n]$ of size L_1 that is continuous in σ , there are at most d bad indices k for which $|\sigma(k) - k| > L_2$.*

Proof. By Markov’s inequality,

$$\Pr_{h \in H} [|\sigma(h) - h| > L_2] \leq \frac{(\sum_{h \in H} |\sigma(h) - h|) / L_1}{L_2} \leq \frac{5c}{L_2} \leq \frac{d}{L_1},$$

where the last inequality follows by choice of c . □

5.2 Algorithm

As mentioned before, we cannot directly use Theorem 3.1 in the noisy comparison model, because we no longer have a natural initial candidate. Instead, we use the algorithm of Braverman and Mossel [6], which constructs the optimal permutation by induction. We summarize their ideas below. Let $S_1 \subset S_2 \subset \dots \subset S_n = [n]$ be arbitrary subsets of $[n]$ such that $|S_k| = k$. With probability $1 - 2n\epsilon$, both conditions in Proposition 5.1 hold simultaneously for all S_k . We iteratively find an optimal order σ_k for S_k , until we reach σ_n .

Suppose that we have σ_{k-1} , and we want to compute an optimal order σ_k . Let $\{a_k\} = S_k \setminus S_{k-1}$; we can try all the k possible locations for inserting a_k into σ_{k-1} .⁶ We will calculate σ_k for each, and keep the best one. Let π_k be the order σ_{k-1} with the insertion of a_k . By induction, π_k is close to the true order on S_k ; therefore by Proposition 5.1 it is also close to an optimal order σ_k . We can now find σ_k using the algorithm for sorting an almost sorted list.

Acknowledgments: We thank Elchanan Mossel for proposing this question. We are also indebted to Elchanan Mossel and Ning Tan for valuable discussions and technical contributions.

References

- [1] Nir Ailon. An active learning algorithm for ranking from pairwise preferences with an almost optimal query complexity. *Journal of Machine Learning Research*, 13:137–164, 2012. [1.4](#)
- [2] Noga Alon. Ranking tournaments. *SIAM J. Discret. Math.*, 20(1):137–142, January 2006. [1.4](#)

⁶This is a slight simplification of the algorithm in [6], which does something more clever than trying all k possible positions.

- [3] Sanjeev Arora, Alan Frieze, and Haim Kaplan. A new rounding procedure for the assignment problem with applications to dense graph arrangement problems. In *Proceedings 37th annual Symposium on Foundations of Computer Science (FOCS)*, pages 21–30. IEEE, 1996. [1.4](#)
- [4] J. Bartholdi, III, C.A. Tovey, and M.A. Trick. Voting schemes for which it can be difficult to tell who won the election. *Social Choice and Welfare*, 6(2):157–165, 1989. [1.4](#)
- [5] Nadja Betzler, Michael R. Fellows, Jiong Guo, Rolf Niedermeier, and Frances A. Rosamond. Fixed-parameter algorithms for Kemeny rankings. *Theor. Comput. Sci.*, 410(45):4554–4570, 2009. [1.4](#)
- [6] Mark Braverman and Elchanan Mossel. Noisy sorting without resampling. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms, (SODA)*, pages 268–276, 2008. [1.2](#), [1.2](#), [1.2](#), [1.3](#), [1.3](#), [1.4](#), [2.2](#), [3](#), [5.2](#), [5.1](#), [5.3](#), [5.2](#), [6](#)
- [7] Mark Braverman and Elchanan Mossel. Sorting from noisy information. *CoRR*, abs/0910.1191, 2009. ([document](#)), [1.2](#), [1.2](#), [3](#), [1.2](#), [1.3](#), [1.3](#), [4](#), [1.4](#), [2.2](#), [4](#), [4.1](#)
- [8] Flavio Chierichetti, Anirban Dasgupta, Ravi Kumar, and Silvio Lattanzi. On reconstructing a hidden permutation. In *Proceeding of Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, (APPROX/RANDOM)*, pages 604–617, 2014. [1.2](#), [3](#), [1.2](#), [1.4](#)
- [9] Irit Dinur and Samuel Safra. On the hardness of approximating minimum vertex cover. *Annals of Mathematics*, 162:2005, 2004. [1.4](#)
- [10] Guy Even, Joseph (Seffi) Naor, Baruch Schieber, and Madhu Sudan. Approximating minimum feedback sets and multi-cuts in directed graphs. *Algorithmica*, 20:151–174, 1998. [1.4](#)
- [11] Uriel Feige, Prabhakar Raghavan, David Peleg, and Eli Upfal. Computing with noisy information. *SIAM J. Comput.*, 23(5):1001–1018, 1994. [1.4](#)
- [12] Alan Frieze and Ravi Kannan. Quick approximation to matrices and applications. *Combinatorica*, 19(2):175–220, 1999. [1.4](#)
- [13] Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. 2nd edition, 1994. [2.4](#)
- [14] Richard M. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, pages 85–103, 1972. [1.4](#)
- [15] Marek Karpinski and Warren Schudy. Approximation schemes for the betweenness problem in tournaments and related ranking problems. In *Proceeding of Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, (APPROX/RANDOM)*, pages 277–288, 2011. [1.4](#)
- [16] Claire Kenyon-Mathieu and Warren Schudy. How to rank with few errors. In *Proceedings of the Thirty-Ninth annual ACM Symposium on Theory of Computing (STOC)*, pages 95–103, 2007. [1.3](#), [1.4](#)

- [17] Subhash Khot and Oded Regev. Vertex cover might be hard to approximate to within $2-\epsilon$. *J. Comput. Syst. Sci.*, 74(3):335–349, 2008. [1.4](#)
- [18] Tyler Lu and Craig Boutilier. Effective sampling and learning for mallows models with pairwise-preference data. *J. Mach. Learn. Res.*, 15(1):3783–3829, 2014. [1.4](#)
- [19] C. L. Mallows. Non-null ranking models. I. *Biometrika*, 44(1/2):pp. 114–130, 1957. [1.1](#)
- [20] Marina Meila, Kapil Phadnis, Arthur Patterson, and Jeff A. Bilmes. Consensus ranking under the exponential model. In *Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 285–294, 2007. [1.4](#)
- [21] Ariel D. Procaccia, Sashank Jakkam Reddi, and Nisarg Shah. A maximum likelihood approach for selecting sets of alternatives. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 695–704, 2012. [1.4](#)
- [22] Narges Simjour. Improved parameterized algorithms for the Kemeny aggregation problem. In *Proc. of Fourth International Workshop on Parameterized and Exact Computation, (IWPEC)*, pages 312–323, 2009. [1.4](#)