BEYOND A SINGLE CONSENSUS: ALGORITHMS FOR DIVERSE RANK AGGREGATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Rank aggregation combines potentially conflicting rankings into a consensus minimizing its total distance to the inputs. Yet many applications require multiple substantially different but equally good consensuses to accommodate heterogeneous needs or information not captured by the aggregation objective. Given an integer k, *diverse rank aggregation* seeks k optimal consensuses maximizing a diversity objective. We study this problem under the *Spearman footrule* distance for three standard objectives: diameter (the case of $k = 2$), sum dispersion, and min dispersion. The only prior theoretical work considered sum dispersion under Kendall–tau distance from a fixed-parameter-complexity perspective [Arrighi et al., IJCAI'21]; we initiate the study of the polynomial-time approximation landscape.

While standard rank aggregation under the Spearman footrule is solvable in polynomial time, the complexity of its diverse variants remained open. We give polynomial-time exact algorithms for diameter and sum dispersion using new doubled- and k-copy matching formulations over the tight-edge assignment graph. In contrast, we show that min dispersion is NP-hard for every fixed $k \geq 3$. Since the notion of min dispersion coincides with diameter for $k = 2$, this yields a sharp complexity dichotomy. For fixed k, we give a randomized factor-2 approximation based on an exact robust-farthest oracle (algorithm): given selected optimal aggregates, it finds another maximizing its minimum distance from them. We design this algorithm using a multivariate determinant, interpolation, and self-reduction, and prove it NP-hard for unbounded k. When k is part of the input, a polynomial-time multiplicative-weights algorithm ensures that every pair has expected distance at least half the optimum, up to any prescribed additive error. We further study diverse solution generation also for near-optimal aggregate rankings.

We complement our theoretical study with empirical results on multiple real-world datasets. Our algorithms attain the optimal sum-dispersion value, perform strongly on min dispersion against natural matching- and solver-based baselines, and furthermore, return rankings that remain highly diverse even under Kendall–tau distance.

1 INTRODUCTION

Rankings are ubiquitous in representing preferences, scores, and recommendations. Rank aggregation asks how to combine several, possibly conflicting, rankings into a single consensus ranking, with applications in social choice and voting (Brandt et al., 2016), web search and metasearch (Dwork et al., 2001a; Aslam & Montague, 2001), database search and classification (Fagin et al., 2003), and group recommendation (Baltrunas et al., 2010). The most common optimization formulation seeks to find a ranking minimizing the sum of its distances to the input set of rankings. Equivalently, it computes a (metric) median of the input rankings; throughout this paper, we refer to this unconstrained median objective as *rank aggregation*.

A single consensus ranking, however, may provide an incomplete view of the preferences expressed by the input rankings. There may be many equally good aggregate rankings, and an arbitrary tie-breaking rule reveals neither the range of optimal decisions nor the sensitivity of the consensus. More generally, an optimization objective is often only a proxy: some considerations are subjective, unavailable at optimization time, or too costly to encode. A small list of diverse optimal solutions lets a human or a downstream procedure choose after seeing the available tradeoffs, and can make the final decision less dependent on unmodeled priorities or changing deployment conditions (Petit & Trapp, 2019; Baste et al., 2022; Hanaka et al., 2023). This motivation is especially natural in information-access systems. Diversified search results can cover different interpretations or aspects of a query (Baste et al., 2022), while topic diversification in recommendation can represent a user's distinct interests and avoid a list of near-redundant alternatives (Ziegler et al., 2005). In rank aggregation, likewise, multiple consensus rankings can have the same optimal aggregation cost

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

or nearly the same cost while ordering the candidates differently. Returning a diverse collection of these rankings reveals the range of optimal or near-optimal outcomes, supports application-specific choice and deliberation, and reduces dependence on arbitrary tie-breaking. Beyond ranking, a similar diverse-solution perspective has been studied for shortest paths and related graph and matroid structures (Hanaka et al., 2022), matchings (Fomin et al., 2024a), minimum cut (de Berg et al., 2023), Boolean satisfiability (Misra et al., 2024; Austrin et al., 2025), longest common subsequences (Shida et al., 2024), string medians (Chakraborty et al., 2026a), among many others.

We study diversity first among optimal aggregate rankings and then under a relaxation to near-optimal rankings, using three standard diversity notions. Perhaps the simplest is the *diameter*, which asks for the two optimal rankings that are farthest apart and has also been considered for diverse matchings, satisfiability, and median selection (Fomin et al., 2024a; Austrin et al., 2025; Chakraborty et al., 2026a). For a collection of k optimal rankings, *sum dispersion* maximizes the sum of the distances over all pairs; this max-sum notion is classical in metric dispersion (Ravi et al., 1994; Cevallos et al., 2016) and has also been studied from a fixed-parameter algorithmic perspective for rank aggregation, albeit under a different distance metric (Arrighi et al., 2021). By contrast, *min dispersion* maximizes the minimum pairwise distance, corresponding to the classical max-min or scatteredness objective (Ravi et al., 1994; Arrighi et al., 2021; Chakraborty et al., 2026a). The former rewards diversity in aggregate, while the latter requires every pair of outputs to be well separated. For $k = 2$, both sum dispersion and min dispersion collapse to the diameter objective. We provide formal definitions in the preliminaries and refer readers to Appendix A for a more elaborate discussion of related work.

We consider one of the classical distance measures on rankings, the *Spearman footrule*, which sums, over all candidates, the absolute difference between their positions in the two rankings—equivalently, it is the ℓ_1 distance between their rank vectors (Spearman, 1904; 1906). The Spearman footrule is simple to evaluate and optimize. It has been used in database similarity search and classification (Fagin et al., 2003), web rank aggregation (Dwork et al., 2001a), metasearch and information retrieval (Aslam & Montague, 2001; Liu, 2009), biomedical rank aggregation (Pihur et al., 2009), and group recommendation (Baltrunas et al., 2010). A further computational advantage is that standard rank aggregation under the Spearman footrule reduces to a *minimum-cost bipartite perfect matching* and is therefore exactly solvable in polynomial time (Dwork et al., 2001a). The rank aggregation under Spearman footrule has also been studied recently under constraints such as fairness (Wei et al., 2022; Chakraborty et al., 2022; 2026b). We contrast this with the closely related *Kendall–tau distance*, which counts pairwise inversions and is widely used for consensus finding (Kemeny, 1959; Brandt et al., 2016; Dwork et al., 2001a). Although the two distances are equivalent up to a factor of two (Diaconis & Graham, 1977), their standard aggregation problems differ sharply: a Spearman-footrule median is computable in polynomial time, whereas aggregation under Kendall–tau is NP-hard (Bartholdi et al., 1989; Dwork et al., 2001a). Prior theoretical work on diverse rank aggregation studies sum dispersion from a fixed-parameter algorithmic perspective (Arrighi et al., 2021), but does not address polynomial-time approximation.

These observations lead to our central question: although a single Spearman-footrule median is computable in polynomial time, how tractable is it to compute a diverse collection of optimal aggregate rankings? We initiate the systematic study of this question from the perspective of polynomial-time approximation algorithms and also consider a relaxation to near-optimal medians. Across standard diversity notions, we give exact algorithms where possible; for min dispersion, we establish a sharp dichotomy—polynomial-time solvability for $k = 2$ and NP-hardness for every fixed $k \geq 3$—and then develop approximation guarantees.

Our contributions. Given a set of n rankings over d candidates and an integer k, we first seek k optimal aggregate rankings that maximize one of the diversity objectives above. Prior theoretical work on diverse Kemeny rank aggregation under Kendall–tau distance provides fixed-parameter tractable algorithms (Arrighi et al., 2021). In contrast, to our knowledge, we are the first to investigate the polynomial-time approximation landscape for diverse rank aggregation, focusing on the Spearman footrule.

Diameter and Sum dispersion. We first design an exact $O(nd + d^3)$-time algorithm for finding two farthest optimal aggregate rankings (Theorem C.1). For that purpose, we construct a doubled copy of the tight-edge assignment graph, in which signed edge weights turn the absolute Spearman footrule distance into a linear matching objective. A canonical-orientation exchange argument and an alternating-cycle decomposition then recover two optimal aggregate rankings from the auxiliary matching. We extend this idea to a collection of k optimal rankings and optimize the sum of all pairwise Spearman footrule distances exactly in $O(nd + (kd)^3)$ time (Theorem 3.1). We introduce a k-copy assignment with layer coefficients $\lambda_t = 2t - k - 1$ for $t \in [k]$; after sorting the k positions assigned to each candidate as $p_1 \leq \cdots \leq p_k$, the identity $\sum_{a<b} |p_a - p_b| = \sum_t \lambda_t p_t$ exactly linearizes all pairwise absolute differences. Collapsing the copies yields a k-regular bipartite multigraph whose factorization into k perfect matchings recovers the desired aggregates; equivalently, the construction is an integral transportation formulation.

Min dispersion. Note that for $k = 2$, both the sum and min dispersion objective coincide (to diameter). For every fixed $k \geq 3$, the decision version is NP-complete even when the input is a set of distinct rankings (Theorem 4.1), and it

admits no fully-polynomial time approximation scheme (FPTAS) unless $\mathrm{P} = \mathrm{NP}$ (Corollary G.10). Together with our exact algorithm for $k = 2$, this establishes a sharp separation from the computational complexity perspective.

Despite this hardness, we obtain approximation guarantees for min dispersion. We design a greedy farthest-first algorithm that starts from an arbitrary optimal aggregate ranking and, at each iteration, adds an optimal aggregate ranking maximizing its minimum distance from those already selected. A packing-versus-covering argument shows that this algorithm achieves a 2-approximation whenever each iteration can be implemented using an exact *robust-farthest oracle* (Theorem 4.2).

Our main algorithmic novelty is an exact robust-farthest oracle for fixed k. We encode the vector of distances from a candidate optimal ranking to the previously selected rankings as the exponent vector of a multivariate determinant. Randomized evaluation and interpolation identify an attainable vector that maximizes its smallest coordinate, after which self-reduction recovers a ranking that realizes that vector. This yields the randomized factor-2 approximation for every fixed k (Theorem 4.2). When k is part of the input, we instead combine multiplicative weight updates with a maximum-weight assignment best-response oracle, obtaining a polynomial-time pairwise ex-ante guarantee: every pair of returned rankings has expected distance at least $(1/2 - \epsilon)$ times the optimal min-dispersion value (Theorem 4.3). Complementing these results, we prove that the exact robust-farthest problem is NP-complete when the number of previously selected rankings is unbounded (Theorem 4.5) via a reduction from the CLOSEST STRING problem.

Approximate aggregate rankings. While generating diverse solutions, considering only the set of optimal aggregate rankings can be too restrictive: a rank-aggregation instance may have a unique optimal aggregate (median) ranking, leaving no nontrivial diversity to recover. In applications, one may instead prefer substantially different consensus rankings while tolerating a small multiplicative loss in aggregation quality. We therefore also study diversity among $(1 + \alpha)$-approximate aggregate rankings (Section 5).

We establish a stark complexity contrast between optimal and approximately optimal aggregate rankings. Although the diameter among optimal footrule aggregates is polynomial-time solvable, finding the diameter among $(1 + \alpha)$-approximate aggregates is NP-complete (Theorem 5.1). Nevertheless, we construct a budgeted assignment oracle—an extension of the algebraic method used for our robust-farthest oracle—that optimizes additive functions over approximately optimal aggregates (Lemma 5.2). Using this oracle, we obtain a randomized polynomial-time $1/2$-approximation for diameter (Theorem J.1), a randomized polynomial-time $(1 - 1/k)$-approximation for sum dispersion (Theorem 5.4), and, for every fixed k, a randomized polynomial-time $1/2$-approximation for min dispersion (Theorem 5.5).

We complement our theoretical results with experiments on real-world ranking data. Our methods attain optimal sum dispersion and perform strongly for min dispersion across the tested datasets: the matching-based algorithm is particularly efficient, while the more computationally intensive MWU algorithm completes on all tested instances. Although designed specifically for the Spearman footrule, our algorithms also perform well under the Kendall–tau distance, retaining meaningful diversity and strong aggregation quality. We also develop a Kendall–tau-specific heuristic that achieves greater diversity, albeit at a higher aggregation cost, revealing a clear quality–diversity tradeoff.

2 PRELIMINARIES

For an integer $d \geq 1$, let $[d]$ denote the set $\{1, \ldots, d\}$. Let $\mathcal{S}_d$ denote the set of all full rankings (permutations) over the candidate set $[d]$. For $\sigma \in S_d$, let $\sigma(i)$ denote the rank/position of candidate i in σ.

Distance metric. The *Spearman footrule* distance between two rankings $\pi, \sigma \in \mathcal{S}_d$ is defined as $F(\pi, \sigma) := \sum_{i \in [d]} |\pi(i) - \sigma(i)|$.

Now, we can formally define the problem of rank aggregation under the Spearman footrule distance.

Definition 2.1 (Rank Aggregation)**.** Given a (multi)set S of rankings over d candidates, the (median) *rank aggregation* problem asks to find a ranking $\sigma \in \mathcal{S}_d$ that minimizes the objective function $\sum_{\pi \in S} F(\pi, \sigma)$.

Throughout the paper, the input set of rankings S is a multiset and may contain duplicate rankings.

For an input set S, let its optimal rank aggregation objective be denoted by $C^\star$, i.e., $C^\star = \min_{\sigma \in \mathcal{S}_d} \sum_{\pi \in S} F(\pi, \sigma)$. Let $\mathcal{O}_S$ denote the set of all rankings of $\mathcal{S}_d$ which are optimal aggregate rankings for S, more specifically, $\mathcal{O}_S = \{\sigma \in \mathcal{S}_d : \sum_{\pi \in S} F(\pi, \sigma) = C^\star\}$. When the set S is clear from the context, for brevity, we will drop the subscript and simply write $\mathcal{O}$. In the rest of the paper, we use n to denote $|S|$, the size of the input set.

The rank aggregation problem under the Spearman footrule distance is known to be solvable in polynomial time via a reduction to the *weighted bipartite matching problem* (Dwork et al., 2001a). The linear-programming formulation of this perfect-matching problem and its dual are provided in Appendix B. With respect to an optimal dual solution,

an edge assigning a candidate to a position is called *tight* when its assignment cost equals the corresponding dual bound. We consider the bipartite graph $G_0 = ([d] \cup [d], E_0)$ with candidates on one side, positions on the other, and the edge-set E_0 is the set of all these tight edges. The tight-edge characterization states that a ranking is an optimal aggregate if and only if its corresponding perfect matching uses only edges of G_0. Formal definitions and the proof are deferred to Appendix B.

Diverse Rank Aggregation. Our goal is to find multiple *optimal* (or often approximate) solutions to the rank aggregation problem, which are as diverse as possible in terms of the Spearman Footrule distance. For diversity, we consider two optimization objectives: maximizing the sum of distances between rankings in the output set, or maximizing the minimum distance between any two rankings in the output set. Formally, the two problems are as follows.

Definition 2.2 (Max-sum Diverse Rank Aggregation)**.** Given a set S of rankings over d candidates and an integer $k \geq 2$, the *max-sum diverse rank aggregation problem* asks to find a set of rankings $T = \{\sigma_1, \sigma_2, \cdots, \sigma_k\}$ such that for all $\sigma_i \in T$, $\sigma_i \in \mathcal{O}_S$, and the set T maximizes the *sum dispersion* $\sum_{1 \leq a < b \leq k} F(\sigma_a, \sigma_b)$.

For a set $T \subseteq \mathcal{O}_S$, let the max-sum diversity objective of T be denoted by $\texttt{Obj}_{sum}(T) := \sum_{1 \leq a < b \leq k} F(\sigma_a, \sigma_b)$.

Definition 2.3 (Max-min Diverse Rank Aggregation)**.** Given a set S of rankings over d candidates and an integer $k \geq 2$, the *max-min diverse rank aggregation problem* asks to find a set of rankings $T = \{\sigma_1, \sigma_2, \cdots, \sigma_k\}$ such that for all $\sigma_i \in T$, $\sigma_i \in \mathcal{O}_S$, and the set T maximizes the *min dispersion* $\min_{1 \leq a < b \leq k} F(\sigma_a, \sigma_b)$.

For a set $T \subseteq \mathcal{O}_S$, let the max-min diversity objective of T be denoted by $\texttt{Obj}_{min}(T) := \min_{1 \leq a < b \leq k} F(\sigma_a, \sigma_b)$. Let $optmin$ denote the optimal objective value for the max-min diversity objective. If $\texttt{Obj}_{min}(T) \geq \alpha \texttt{OPT}_{min}$ (for some $\alpha \in (0, 1]$), we call T an α-approximate solution for the max-min diversity objective (for optimal aggregate ranking set $\mathcal{O}_S$ with respect to input S).

Note that when $k = 2$, both optimization problems are identical and are referred to as *diameter* in the literature. We allow T to be a multiset, so the problem is well-defined even if fewer than k optimal solutions exist for the given input S. These definitions extend naturally to approximate optimal aggregate rankings, which we discuss in Section 5.

3 Exact Algorithm for Max-Sum Dispersion

In this section, we present an exact algorithm for max-sum diverse rank aggregation for arbitrary $k \geq 2$. The special case $k = 2$ is precisely the diameter objective; its exact algorithm and a simpler analysis are presented in Appendix C. Our main result for sum dispersion is the following.

Theorem 3.1. *There exists an algorithm that, given as input a set of rankings S over d candidates and an integer $k \geq 2$, solves the max-sum Diverse Rank Aggregation problem exactly. The algorithm runs in time $O(nd + (kd)^3)$.*

Algorithm description. We first solve the assignment formulation of rank aggregation and construct the tight-edge graph G_0 with edge set E_0 (for the details, see Appendix B). By the tight-edge characterization (Lemma B.1), every perfect matching of G_0 corresponds to an optimal aggregate ranking, so restricting the construction to E_0 ensures that every output ranking is optimal.

To optimize k rankings simultaneously, we lift G_0 to a bipartite graph H_k containing k labeled copies of every candidate and k copies of every position. For $t \in \{1, \ldots, k\}$, define $\lambda_t := 2t - k - 1$, so that $\lambda_1 < \lambda_2 < \cdots < \lambda_k$. A maximum-weight perfect matching in H_k selects k admissible positions for each candidate while using each position exactly k times overall. The coefficient λ_t is assigned to the tth candidate copy; because these coefficients are increasing, an exchange argument orders the selected positions as $p_1 \leq \cdots \leq p_k$. The identity $\sum_t \lambda_t p_t = \sum_{a<b} |p_a - p_b|$ then makes the matching weight equal to that candidate's contribution to the total pairwise footrule distance. Finally, merging the copies produces a k-regular bipartite multigraph. Decomposing it into k perfect matchings gives k matchings of G_0, and hence the desired k optimal aggregate rankings.

Formally, let the left vertex set of H_k contain k copies of each candidate and the right vertex set contain k copies of each position. For every edge $(i, p) \in E_0$, connect every copy of candidate i to every copy of position p. Label the copies of candidate i by $i^1, i^2, \ldots, i^k$, and assign weight $\lambda_t p$ to every edge from i^t to a copy of position p.

The correctness proof uses the identity induced by the coefficients λ_t, an exchange argument that orders each candidate's copies by rank, and a perfect-matching factorization of the resulting regular bipartite multigraph. Together, these show that the matching weight equals sum dispersion. The complete correctness proof and running-time analysis are deferred to Appendix D.

Algorithm 1 MAXSUMDIVERSITY(S)

1: Reduce rank aggregation on S to min-cost perfect matching on graph G.
2: Solve min-cost perfect matching on G, obtaining an optimal primal-dual solution.
3: Construct tight edge graph G_0 with edge set E_0.
4: Construct bipartite graph H_k with k copies of candidates on the left and k copies of each rank on the right, with edge set and edge weights as specified in the text.
5: $N^\star \leftarrow$ maximum-weight perfect matching on graph H_k.
6: Decompose $N^\star$ into perfect matchings $M_1, M_2, \ldots, M_k$ from candidates to ranks.
7: For each $i \in [k]$, $\sigma_i \leftarrow$ ranking corresponding to M_i.
8: **return** $T = \{\sigma_1, \sigma_2, \ldots, \sigma_k\}$.

4 RESULTS FOR MAX-MIN DIVERSITY

In this section, we study the problem of max-min diverse rank aggregation. We first show that the problem is NP-hard even for any $k \geq 3$. Since max-sum and max-min rank aggregation coincide when $k = 2$, our hardness result together with the exact algorithm presented in Section 3 provides a sharp contrast in computational complexity between $k = 2$ and $k \geq 3$. Next, we design a 2-approximation algorithm for fixed k that runs in $nd + d^{O(k)}$ time.

Hardness of Max-Min Diverse Rank Aggregation. We now prove a hardness result for max-min diverse rank aggregation. For a fixed integer k, the decision problem receives a set $S \subseteq \mathcal{S}_d$ and an integer D, and asks whether there are $\sigma_1, \ldots, \sigma_k \in \mathcal{O}_S$ such that for all $1 \leq a < b \leq k$, $F(\sigma_a, \sigma_b) \geq D$.

Theorem 4.1 (NP-completeness for every fixed $k \geq 3$)**.** *For every fixed integer $k \geq 3$, the decision version of max-min diverse rank aggregation is NP-complete.*

To show the above hardness result, we start with the edge-precoloring extension problem and create a weighted bipartite graph by embedding several bi-cliques and adjusting edge weights to encode it as k diverse perfect matchings (where distance between two matchings is measured by the symmetric difference of the edge sets involved in the matchings). A threshold-preserving path gadget converts weighted symmetric-difference distance into Spearman footrule distance while keeping the lower-order displacement below one scaled unit. An inverse-Laplacian construction then realizes exactly the lifted matchings as optimal aggregates. Finally, we append the same dummy-candidate suffix to every input ranking, swapping a different adjacent pair in each ranking; this makes the inputs distinct while forcing every optimal aggregate to use the canonical suffix, so pairwise distances between optimal aggregates remain unchanged. We defer the full construction and proof to Appendix G.

Approximation Algorithms for Max-Min Diversity.

Theorem 4.2. *There exists a randomized algorithm that, given as input a set of rankings S over d candidates, an integer $k \geq 2$ and a $0 < \zeta < 1$, outputs a $T \subseteq \mathcal{O}_S$ where $|T| = k$ such that $\mathtt{Obj}_{min}(T) \geq \mathtt{OPT}_{min}/2$ with probability at least $1 - \zeta$. The running time of the algorithm is $O(nd) + d^{O(k)}\,\mathrm{poly}(\log(1/\zeta))$.*

We note that the failure probability of the above algorithm is over the approximation guarantee, and the algorithm always returns rankings in $\mathcal{O}_S$. Next, we give an algorithm that, in expectation, provides a $(2+\epsilon)$-approximation (formally, it is not a worst-case approximation but an ex-ante guarantee) and runs in polynomial time, where k is part of the input.

Theorem 4.3. *There exists a randomized polynomial-time algorithm that, given as input a set of rankings S over d candidates, an integer k, and a $0 < \epsilon < 1/2$, outputs $T = \{\sigma_1, \ldots, \sigma_k\} \subseteq \mathcal{O}_S$ such that*

$$\min_{1 \leq a < b \leq k} \mathbb{E}[F(\sigma_a, \sigma_b)] \geq \left(\frac{1}{2} - \epsilon\right) \mathtt{OPT}_{min},$$

where the expectation is over the algorithm's randomness.

Before describing our algorithmic results, we introduce a few key definitions. For a nonempty set $T = (\sigma_1, \ldots, \sigma_t) \subseteq \mathcal{O}_S$, define its *robust-farthest radius* by $R(T) := \max_{\tau \in \mathcal{O}_S} \min_{\sigma_j \in T} F(\tau, \sigma_j)$.

Definition 4.4 (α-robust-farthest oracle)**.** For $0 < \alpha \leq 1$, an *α-robust-farthest oracle* takes as input a set $T \subseteq \mathcal{O}_S$ and returns a ranking $\tau \in \mathcal{O}_S$ satisfying $\min_{\sigma \in T} F(\tau, \sigma) \geq \alpha R(T)$.

Description of the algorithm framework. Suppose we have an α-robust farthest oracle. Then the algorithm is as follows. We iteratively construct the output ranking set T as follows. First, find any optimal aggregate ranking $\sigma_1 \in \mathcal{O}_S$

and let $T_1 = \{\sigma_1\}$. For $i = 2 \ldots, k$, we proceed greedily. Run the oracle with T_{i-1} as input, obtain a ranking $\sigma_i \in \mathcal{O}_S$, and let $T_i = T_{i-1} \cup \{\sigma_i\}$. Finally, output the set T_k as T. The main difficulty comes from designing this oracle.

Algorithm 2 FARTHEST-FIRST(S, k)

```
σ_1 ← any optimal aggregate ranking of S.
T_1 = {σ_1}
for i = 2 ... k do
    σ_i ← α-robust-farthest-oracle(T_{i-1})
    T_i = T_{i-1} ∪ {σ_i}
return T_k
```

The farthest-first analysis shows that an α-robust-farthest oracle yields an $\alpha/2$-approximation for max-min diversity.

Construction of approximate robust-farthest oracle. The main novelty lies in designing an α-robust-farthest oracle for $\alpha = 1$. For fixed k, we construct a randomized exact oracle using a novel algebraic method as follows: We first consider a multivariate determinant that encodes every distance vector attainable by an optimal aggregate ranking. Then we use interpolation to identify a vector whose minimum coordinate equals the robust-farthest radius $R(T)$. Finally, we use self-reduction to recover a ranking attaining it. We provide the details in Appendix E.1. Each of the $k-1$ oracle calls has failure probability $\zeta/(k-1)$, and then applying a union bound yields the randomized factor-two guarantee in Theorem 4.2.

For arbitrary k, a maximum-weight assignment gives an exact best response to every weighted combination of the current centers. A multiplicative weight update algorithm then returns a randomized ranking τ satisfying $\min_j \mathbb{E}[F(\tau, \sigma_j)] \geq R(T) - \eta$. Iterating this response yields $\min_{a<b} \mathbb{E}[F(\sigma_a, \sigma_b)] \geq \texttt{OPT}_{min}/2 - \eta$, which, after scaling the additive error, implies Theorem 4.3. The full algorithm and analysis are deferred to Appendix E.2.

Hardness of the Exact Robust-Farthest Query. It is natural to ask whether an exact robust-farthest oracle can be constructed in polynomial time. We show that this is unlikely, as the exact robust-farthest query is NP-hard when the size of T is arbitrary (i.e., not constant).

Theorem 4.5. *Given a set S of rankings over d candidates, rankings $T = \{\sigma_1, \ldots, \sigma_t\} \subseteq \mathcal{O}_S$ and an integer D, the problem of determining whether there exists $\tau \in \mathcal{O}_S$ such that $\min_{\sigma_j \in T} F(\tau, \sigma_j) \geq D$, is NP-complete.*

We establish the above hardness result by reducing from the *closest string problem* (the center problem under Hamming distance), which is known to be NP-hard. The proof encodes every bit by the order of an adjacent candidate pair, so the Spearman footrule distance becomes exactly twice the Hamming distance. Complementing a candidate string then converts the closest-string requirement into a robust-farthest requirement. We defer the complete reduction and correctness proof to Appendix F.

5 Diversity Among Approximate Aggregate Rankings

So far, we have required every output ranking to be an exact median (optimal aggregate ranking) of the input rankings. We now relax this requirement and ask for diverse rankings whose aggregation costs are each within a factor $1 + \alpha$ of optimum. This relaxation enlarges the feasible family and can reveal alternatives that are unavailable among exact medians, but it also changes the computational landscape: even the diameter variant (i.e., $k = 2$) becomes NP-hard. Nevertheless, we obtain polynomial-time approximation algorithms for diameter and sum dispersion, and for min dispersion when the requested number of rankings (k) is fixed.

For a set S of n rankings over d candidates, recall that we use $C^\star$ to denote the optimal rank aggregation objective. Let $B_d := \lfloor d^2/2 \rfloor$ be the maximum Spearman footrule distance between two rankings. For some $\alpha \geq 0$, let $\mathcal{A}_\alpha(S)$ (for brevity, simply $\mathcal{A}_\alpha$) be the set of all $(1+\alpha)$-approximate aggregate rankings of S, i.e., $\mathcal{A}_\alpha := \{\sigma \in \mathcal{S}_d : \sum_{\pi \in S} F(\sigma, \pi) \leq (1+\alpha)C^\star\}$. Now, we consider the problem of finding diverse rankings in $\mathcal{A}_\alpha$. We will use similar notation as before. We write the optimal diameter objective as $\texttt{OPT}^\alpha_{\text{diam}} := \max_{\sigma,\tau \in \mathcal{A}_\alpha} F(\sigma, \tau)$, the optimal max-sum diversity objective as $\texttt{OPT}^\alpha_{\text{sum}} := \max_{T \subseteq \mathcal{A}_\alpha} \sum_{\sigma_a, \sigma_b \in T} F(\sigma_a, \sigma_b)$, and the optimal max-min diversity objective as $\texttt{OPT}^\alpha_{\text{min}} := \max_{T \subseteq \mathcal{A}_\alpha} \min_{\sigma_a, \sigma_b \in T} F(\sigma_a, \sigma_b)$.

Diverse approximately optimal solutions under individual quality guarantees have also been studied in broader combinatorial settings (Gao et al., 2022; Hanaka et al., 2023).

Hardness Already for Diameter. For the exact medians, the diameter problem is polynomial-time solvable (Theorem C.1). This tractability disappears when the feasible rankings are allowed to be small approximate medians.

Theorem 5.1 (Hardness of approximate-median diameter)**.** *Given a set S of distinct rankings, a rational $0 < \alpha < 1$, and an integer L, the problem of deciding whether there exist $\sigma, \tau \in \mathcal{A}_\alpha$ such that $F(\sigma, \tau) \geq L$, is NP-complete.*

To prove the above result, we start with the reduction that establishes Theorem 4.1. We then combine balanced perfect-matching factorization, a path gadget that translates weighted symmetric difference into Spearman footrule distance, and an exact realization of the approximate-median budget to obtain the desired hardness result; see Appendix H.

Approximation Algorithms for Diversity. Our approximation algorithms rely on the following oracle, which serves as the main tool for optimizing the different diversity objectives over approximate medians. This result extends Lemma E.5.

Lemma 5.2. *Let $0 < \delta < 1$. For every candidate $i \in [d]$ and position $p \in [d]$, let $w^j_{i,p}$ ($j \in [t]$) be a given non-negative integer label value. For any $\sigma \in \mathcal{A}_\alpha$, define the vector*

$$\boldsymbol{w}(\sigma) := \left(\sum_{i=1}^{d} w^1_{i,\sigma(i)}, \ldots, \sum_{i=1}^{d} w^t_{i,\sigma(i)} \right).$$

Denote W_j as the maximum value of any coordinate of $\boldsymbol{w}$. Let $\Phi : \prod_{j=1}^{t} \{0, \ldots, W_j\} \longrightarrow \mathbb{Q}$ be any polynomial-time computable function. There exists a randomized algorithm that returns a ranking $\widehat{\sigma}$ where $\widehat{\sigma} \in \mathcal{A}_\alpha$ and, with probability at least $1 - \delta$, $\Phi\big(w(\widehat{\sigma})\big) = \max_{\sigma \in \mathcal{A}_\alpha} \Phi\big(w(\sigma)\big)$ and runs in time polynomial in d, t, $(nB_d + 1)(W_j + 1)^t$ and $\log(1/\delta)$.

We again use the algebraic method similar to that we use to build our robust-farthest oracle (used in the proof of Theorem 4.2). Our approach encodes aggregation cost and labels as determinant exponents, then interpolates attainable vectors, and finally recovers an optimizer by self-reduction; see Appendix I.

Theorem 5.4. *There exists a randomized polynomial-time algorithm that, given a set of rankings S, an $\alpha \geq 0$, an integer $k \geq 2$, and a $0 < \zeta < 1$, returns a list of rankings T from $\mathcal{A}_\alpha$ satisfying $\mathtt{Obj}_{sum}(T) \geq (1 - 1/k)\mathtt{OPT}^{\alpha}_{\text{sum}}$ with probability at least $1 - \zeta$.*

Note that the failure probability appears only in the approximation guarantee. For the special case $k = 2$, sum dispersion is exactly the diameter objective, so the same $1/2$-approximation follows directly. A direct statement and proof are given in Appendix J. To prove the result for the general k-max-sum-diversity objective, we use a threshold-coordinate decomposition to show that the Spearman footrule distance is of negative type, which allows us to apply a conditional-gradient construction and conditional-expectation rounding; see Appendix K.

Theorem 5.5. *There exists a randomized algorithm that, given a set of rankings S, an $\alpha \geq 0$, an integer $k \geq 2$, and a $0 < \zeta < 1$, returns a list of rankings T from $\mathcal{A}_\alpha$ satisfying $\mathtt{Obj}_{min}(T) \geq \frac{1}{2}\mathtt{OPT}^{\alpha}_{\min}$ with probability at least $1 - \zeta$. The running time has an additional factor $(B_d + 1)^{O(k)}$.*

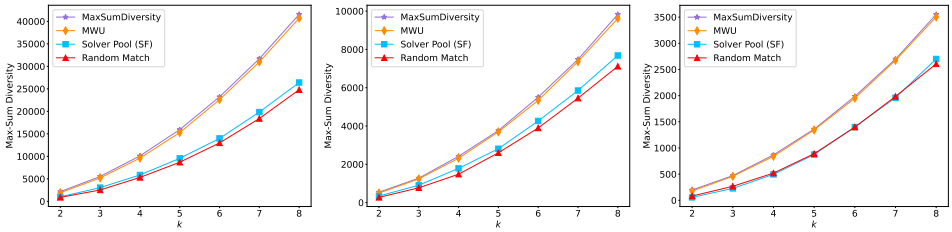
Note that the failure probability appears only in the approximation guarantee. Further, for any fixed k, the above algorithm runs in polynomial time. The argument is similar to that of Theorem 4.2, but we use the oracle of Lemma 5.2 instead of the robust-farthest oracle. The full proof is given in Appendix L.

6 Experiments

We implement our algorithms and several natural baselines, and we evaluate their empirical performance and scalability on real-world datasets. Our code and the exact datasets are available in the supplementary material.

Hardware and software configurations. We use a machine running Ubuntu 24.04, with code run on a machine with an AMD EPYC 7763 and allowed to use 16 threads and 128GB of RAM. The implementation is run in Python 3.13.

Baselines. We consider several baseline algorithms for comparison. First, Random Match is a simple baseline: we compute the tight edge graph and corresponding primal-dual solution, as described in Appendix B, and then find a random perfect matching via assigning edges weights draw uniformly at random. This guarantees all rankings returned are optimal aggregate rankings, but has no guarantee on diversity quality. Next, Solver Pool iteratively builds a solution set by reducing the problem to minimum-cost bipartite perfect matching. At each iteration, we add constraints to prevent any ranking in the solution set from being a feasible solution, to avoid this heuristic from repeatedly adding the same ranking to the solution. We also implement a program using OR-Tools with the CP-SAT solver (Perron & Didier, 2025) to find an optimal solution to the max-sum diverse rank aggregation and max-min diverse rank aggregation, with a timeout of 2 hours. This lets us verify how close our solutions are to optimal, or near-optimal if timeout occurs.



(a) Movielens dataset (b) University dataset (2015) (c) Country dataset (2014)

Figure 1: Performance of algorithms on max-sum diverse rank aggregation

For the Kendall tau distance study, we implement the approximation algorithm for rank aggregation under Kendall tau distance introduced by Ailon et al. (2008), which runs Kwiksort and Pick-A-Perm and returns the better of the two as the aggregate ranking; we label this algorithm KWIKSORT HEURISTIC. We run it k times, use the k returned rankings as the solution set T, and do not know any bound on the diversity quality of the returned rankings.

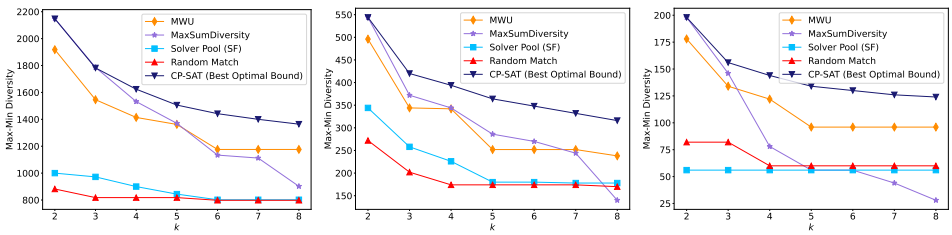
Datasets. We use the movielens dataset which has been used previously to study the related task of fairness in rank aggregation (Wei et al., 2022; Chakraborty et al., 2025; 2026b). This dataset consists of 7 input rankings over 268 candidates. We also use the university dataset and country dataset provided by Boehmer & Schaar (2023). We use the 2015 university dataset with 15 input rankings over 200 candidates, and the 2014 country dataset with 15 input rankings over 115 candidates, as these are among the larger dataset instances available. Our experiments show all these datasets indeed have a diverse set of optimal aggregate rankings.

Research Questions. We implement Algorithm 1 as MAXSUMDIVERSITY and Algorithm 2 that uses a multiplicative weight update-based robust farthest oracle (see Algorithm 4) as MWU. The parameter setting used for MWU (see Theorem 4.3) is $\epsilon = 0.05$. We consider the following research questions in performing the experiments.

- RQ1: How good are the solutions our algorithms produce for the max-sum diverse rank aggregation problem? We expect MAXSUMDIVERSITY to produce an optimal solution, as proven in our analysis, and we seek to confirm this empirically. It would also be interesting to study whether an algorithm like MWU, which is designed to optimize minimum diversity, still finds a diverse output set under the max-sum objective.
- RQ2: What is the quality of the solutions produced by our algorithms for the max-min diverse rank aggregation problem? In our theoretical analysis, we have shown that MWU is effectively a $(2+\epsilon)$-approximation (ex-ante) algorithm in expectation for any $\epsilon > 0$. How well does this algorithm perform in practice? Further, do algorithms that optimize the max-sum objective, such as MAXSUMDIVERSITY, find rankings that are also of good quality for the max-min objective?
- RQ3: How well do our algorithms scale to real-world instances? Are our algorithms much more efficient than implementations using exact solvers?
- RQ4: Do our proposed algorithms find robust solutions which also achieve good performance for diverse rank aggregation under the closely related Kendall tau distance metric? The Kendall tau distance between two rankings is equal to the number of pairs of candidates where the two rankings disagree on the relative order of the pairs. For any two rankings, the Spearman footrule distance and the Kendall tau distance are bounded by a factor of two (Diaconis & Graham, 1977). Therefore, diverse rankings in the Spearman footrule distance should also be diverse under the Kendall tau distance. However, while our algorithms are guaranteed to produce optimal aggregate rankings under the Spearman footrule distance, they are only guaranteed to be 2-approximation aggregate rankings under the Kendall tau distance. Therefore, we study whether our algorithm finds rankings that are both good aggregate rankings and diverse rankings under the Kendall tau distance.

Results.

RQ1 - Performance for max-sum diverse rank aggregation. We plot the results of the algorithms as k is increased on the three datasets in figures 1a, 1b and 1c. The empirical results confirm that MAXSUMDIVERSITY produces an optimal solution, with the max-sum objective of its output solution equal to the optimal value found by CP-SAT. In fact, on the larger movielens dataset, the solution returned by CP-SAT within the timeout has a lower objective value than that of MAXSUMDIVERSITY. This is due to CP-SAT returning the best feasible solution it has found after reaching the timeout. It is unable to find the optimal solution within two hours, showing the strength of our matching-based algorithm compared to a generic exact formulation. We also observe that MWU output rankings achieve near-optimal



(a) Movielens dataset (b) University dataset (2015) (c) Country dataset (2014)

Figure 2: Performance of algorithms on max-min diverse rank aggregation

sum diversity, despite the lack of theoretical guarantees. This suggests that an iterative greedy approach tends to work well in real-world datasets despite the lack of optimality guarantees. In comparison, the simpler baseline SOLVER POOL yields much poorer results, as expected, because it only adds constraints to avoid finding the same aggregate ranking and can therefore find another optimal aggregate very close to the existing pool of rankings.

RQ2 - Performance for max-min diverse rank aggregation. We next turn our attention to study the performance on max-min diverse rank aggregation, with the results as k is increased plotted in figures 2a, 2b and 2c. We observe that MWU tends to perform much better than the other algorithms, and in general, performs better than the $(2+\epsilon)$-approximation (ex-ante) of theory compared to the results of CP-SAT. The rankings returned by MAXSUMDIVERSITY show interesting results for the max-min objective. In particular, it sometimes performs extremely poorly and can even find an output with a better max-min objective when k increases, contrary to the intuition that the max-min objective should decrease as k increases. This is not unexpected, as it optimizes the max-sum objective and thus offers no guarantee on the max-min objective; for some k, it may pick two rankings that are close to achieving a larger sum of pairwise distances over the full set of output rankings. Therefore, in practice, an algorithm designed for the max-min solution, like MWU, is indeed necessary to achieve good results. Lastly, we observe that the simple (and natural) baselines SOLVER POOL and RANDOM MATCH do not perform well on the max-min objective. In particular, even at small values of k, the baselines can find two close optimal aggregate rankings as part of their solution, yielding a poor-quality solution for the max-min objective.

RQ3 - Scalability. Our algorithms scale well across all three datasets: all proposed methods except MWU complete very quickly, while MWU remains practical even on the largest instance; in contrast, the generic CP-SAT formulation frequently reaches the two-hour timeout. We provide running-time plots with additional details in Appendix M.

RQ4 - Robustness to Kendall tau distance. The rankings returned by our algorithms remain diverse under Kendall–tau distance, with performance trends broadly resembling those under the Spearman footrule. Although KWIKSORT HEURISTIC achieves greater Kendall–tau diversity, its rankings have substantially higher Kendall–tau aggregation cost, indicating a clear tradeoff between aggregation quality and diversity; see Appendix M for details.

7 CONCLUSION

We initiate the study of the polynomial-time approximation landscape for diverse rank aggregation under the Spearman footrule. We give polynomial-time exact algorithms for diameter and sum dispersion through new matching formulations, while establishing a sharp complexity transition for min dispersion: the problem is polynomial-time solvable for $k = 2$ but NP-complete for every fixed $k \geq 3$. For min dispersion, we develop a randomized factor-two approximation for fixed k using an algebraic robust-farthest oracle, and a polynomial-time multiplicative-weights algorithm with a pairwise ex-ante guarantee for arbitrary k. We further show that allowing $(1+\alpha)$-approximate medians fundamentally changes the complexity landscape: even diameter becomes NP-complete, although all three diversity objectives admit meaningful approximation guarantees. Our experiments complement these results by demonstrating the scalability and empirical effectiveness of the proposed algorithms on real-world ranking data.

It would be interesting to obtain a polynomial-time guarantee on the realized min dispersion for arbitrary k, or to establish a corresponding inapproximability barrier. Other intriguing questions include derandomizing and improving the running time of the algebraic robust-farthest oracle, and determining whether the approximation guarantees for approximate medians are tight. Finally, building theoretical understanding for Kendall–tau distance, partial or tied rankings, and rank aggregation with fairness or other application-specific constraints could broaden understanding of the tradeoff between consensus quality and diversity.

REFERENCES

Nir Ailon, Moses Charikar, and Alantha Newman. Aggregating inconsistent information: ranking and clustering. *Journal of the ACM (JACM)*, 55(5):1–27, 2008.

Juan A. Aledo, José A. Gámez, and Alejandro Rosete. A consensus set for the aggregation of partial rankings: The case of the optimal set of bucket orders problem. *Information Fusion*, 124:103353, 2025. doi: 10.1016/j.inffus.2025.103353. URL https://doi.org/10.1016/j.inffus.2025.103353.

Emmanuel Arrighi, Henning Fernau, Daniel Lokshtanov, Mateus de Oliveira Oliveira, and Petra Wolf. Diversity in kemeny rank aggregation: A parameterized approach. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pp. 10–16, 2021. doi: 10.24963/ijcai.2021/2. URL https://doi.org/10.24963/ijcai.2021/2.

Javed A Aslam and Mark Montague. Models for metasearch. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 276–284, 2001.

Per Austrin, Ioana O Bercea, Mayank Goswami, Nutan Limaye, and Adarsh Srinivasan. Algorithms for the diverse-k-sat problem: the geometry of satisfying assignments. In *52nd International Colloquium on Automata, Languages, and Programming (ICALP 2025)*, pp. 14–1. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2025.

Linas Baltrunas, Tadas Makcinskas, and Francesco Ricci. Group recommendations with rank aggregation and collaborative filtering. In *Proceedings of the Fourth ACM Conference on Recommender Systems*, pp. 119–126, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-906-0. doi: 10.1145/1864708.1864733.

J. Bartholdi, 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, Apr 1989. ISSN 1432-217X.

Julien Baste, Michael R Fellows, Lars Jaffke, Tomáš Masařík, Mateus de Oliveira Oliveira, Geevarghese Philip, and Frances A Rosamond. Diversity of solutions: An exploration through the lens of fixed-parameter tractability theory. *Artificial Intelligence*, 303:103644, 2022. doi: 10.1016/j.artint.2021.103644. URL https://doi.org/10.1016/j.artint.2021.103644.

Niclas Boehmer and Nathan Schaar. Collecting, classifying, analyzing, and using real-world ranking data. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '23, pp. 1706–1715, Richland, SC, 2023. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450394321.

Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D Procaccia. *Handbook of computational social choice*. Cambridge University Press, 2016.

Alfonso Cevallos, Friedrich Eisenbrand, and Rico Zenklusen. Max-sum diversity via convex programming. In *32nd International Symposium on Computational Geometry (SoCG 2016)*, volume 51 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 26:1–26:14. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2016. doi: 10.4230/LIPIcs.SoCG.2016.26. URL https://doi.org/10.4230/LIPIcs.SoCG.2016.26.

Diptarka Chakraborty, Syamantak Das, Arindam Khan, and Aditya Subramanian. Fair rank aggregation. *Advances in Neural Information Processing Systems*, 35:23965–23978, 2022. Full version: arXiv preprint arXiv:2308.10499.

Diptarka Chakraborty, Himika Das, Sanjana Dey, and Alvin Hong Yao Yan. Improved rank aggregation under fairness constraint. In James Kwok (ed.), *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pp. 330–338. International Joint Conferences on Artificial Intelligence Organization, 8 2025. Main Track.

Diptarka Chakraborty, Rudrayan Kundu, Nidhi Purohit, and Aravinda Kanchana Ruwanpathirana. Maximizing Diversity in (Near-)Median String Selection. In Philip Bille and Nicola Prezza (eds.), *37th Annual Symposium on Combinatorial Pattern Matching (CPM 2026)*, volume 369 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 12:1–12:15, Dagstuhl, Germany, 2026a. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. ISBN 978-3-95977-420-8. doi: 10.4230/LIPIcs.CPM.2026.12. URL https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.CPM.2026.12.

Diptarka Chakraborty, Arya Mazumdar, Barna Saha, and Alvin Hong Yao Yan. Fairness in aggregation: Optimal top-k and improved full ranking. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*. PMLR, 2026b. https://arxiv.org/abs/2605.23265.

Mark de Berg, Andrés López Martínez, and Frits C. R. Spieksma. Finding diverse minimum s-t cuts. In Satoru Iwata and Naonori Kakimura (eds.), *34th International Symposium on Algorithms and Computation, ISAAC 2023, Kyoto, Japan, December 3-6, 2023*, volume 283 of *LIPIcs*, pp. 24:1–24:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023. doi: 10.4230/LIPICS.ISAAC.2023.24. URL https://doi.org/10.4230/LIPIcs.ISAAC.2023.24.

Persi Diaconis and Ronald L. Graham. Spearman's footrule as a measure of disarray. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(2):262–268, 1977. https://doi.org/10.1111/j.2517-6161.1977.tb01624.x.

Cynthia Dwork, Ravi Kumar, Moni Naor, and D. Sivakumar. Rank aggregation methods for the web. In *Proceedings of the Tenth International World Wide Web Conference, WWW 10*, pp. 613–622, 2001a. doi: 10.1145/371920.372165.

Cynthia Dwork, Ravi Kumar, Moni Naor, and D Sivakumar. Rank aggregation revisited, 2001b.

Ronald Fagin, Ravi Kumar, and D. Sivakumar. Efficient similarity search and classification via rank aggregation. In *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, San Diego, California, USA, June 9-12, 2003*, pp. 301–312, 2003. doi: 10.1145/872757.872795.

Jiří Fiala. Np completeness of the edge precoloring extension problem on bipartite graphs. *J. Graph Theory*, 43(2): 156–160, June 2003. ISSN 0364-9024.

Fedor V Fomin, Petr A Golovach, Lars Jaffke, Geevarghese Philip, and Danil Sagunov. Diverse pairs of matchings. *Algorithmica*, 86(6):2026–2040, 2024a. doi: 10.1007/s00453-024-01214-7. URL https://doi.org/10.1007/s00453-024-01214-7.

Fedor V Fomin, Petr A Golovach, Fahad Panolan, Geevarghese Philip, and Saket Saurabh. Diverse collections in matroids and graphs. *Mathematical Programming*, 204(1):415–447, 2024b.

Moti Frances and Ami Litman. On covering problems of codes. *Theory of Computing Systems*, 30(2):113–119, 1997. doi: 10.1007/s002240000044.

Jie Gao, Mayank Goswami, C. S. Karthik, Meng-Tsung Tsai, Shih-Yu Tsai, and Hao-Tsung Yang. Obtaining approximately optimal and diverse solutions via dispersion. In *LATIN 2022: Theoretical Informatics: 15th Latin American Symposium, Guanajuato, Mexico, November 7–11, 2022, Proceedings*, pp. 222–239, Berlin, Heidelberg, 2022. Springer-Verlag. ISBN 978-3-031-20623-8. doi: 10.1007/978-3-031-20624-5_14. URL https://doi.org/10.1007/978-3-031-20624-5_14.

Tesshu Hanaka, Yasuaki Kobayashi, Kazuhiro Kurita, See Woo Lee, and Yota Otachi. Computing diverse shortest paths efficiently: A theoretical and experimental study. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 3758–3766, 2022. doi: 10.1609/aaai.v36i4.20290. URL https://doi.org/10.1609/aaai.v36i4.20290.

Tesshu Hanaka, Masashi Kiyomi, Yasuaki Kobayashi, Yusuke Kobayashi, Kazuhiro Kurita, and Yota Otachi. A framework to design approximation algorithms for finding diverse solutions in combinatorial problems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 3968–3976, 2023. doi: 10.1609/aaai.v37i4.25511. URL https://doi.org/10.1609/aaai.v37i4.25511.

Refael Hassin, Shlomi Rubinstein, and Arie Tamir. Approximation algorithms for maximum dispersion. *Operations Research Letters*, 21(3):133–137, 1997. ISSN 0167-6377. doi: https://doi.org/10.1016/S0167-6377(97)00034-5. URL https://www.sciencedirect.com/science/article/pii/S0167637797000345.

John G Kemeny. Mathematics without numbers. *Daedalus*, 88(4):577–591, 1959.

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*, pp. 95–103, New York, NY, USA, 2007. Association for Computing Machinery. ISBN 9781595936318. doi: 10.1145/1250790.1250806. URL https://doi.org/10.1145/1250790.1250806.

Caitlin Kuhlman and Elke Rundensteiner. Rank aggregation algorithms for fair consensus. *Proc. VLDB Endow.*, 13 (12):2706–2719, July 2020. ISSN 2150-8097. doi: 10.14778/3407790.3407855. URL https://doi.org/10.14778/3407790.3407855.

Tie-Yan Liu. Learning to rank for information retrieval. *Found. Trends Inf. Retr.*, 3(3):225–331, March 2009. ISSN 1554-0669. doi: 10.1561/1500000016. URL https://doi.org/10.1561/1500000016.

Neeldhara Misra, Harshil Mittal, and Ashutosh Rai. On the parameterized complexity of diverse SAT. In *35th International Symposium on Algorithms and Computation (ISAAC 2024)*, volume 322 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 50:1–50:18. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2024. doi: 10.4230/LIPIcs.ISAAC.2024.50. URL https://doi.org/10.4230/LIPIcs.ISAAC.2024.50.

Laurent Perron and Frédéric Didier. Cp-sat, 2025. URL https://developers.google.com/optimization/cp/cp_solver/.

Thierry Petit and Andrew C Trapp. Enriching solutions to combinatorial problems via solution engineering. *INFORMS Journal on Computing*, 31(3):429–444, 2019. doi: 10.1287/ijoc.2018.0855. URL https://doi.org/10.1287/ijoc.2018.0855.

Vasyl Pihur, Susmita Datta, and Somnath Datta. RankAggreg, an R package for weighted rank aggregation. *BMC bioinformatics*, 10(1):62, 2009.

S. S. Ravi, Daniel J. Rosenkrantz, and Giri Kumar Tayi. Heuristic and special case algorithms for dispersion problems. *Operations Research*, 42(2):299–310, 1994. doi: 10.1287/opre.42.2.299.

Jacob T. Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *Journal of the ACM*, 27(4): 701–717, 1980. doi: 10.1145/322217.322225.

Yuto Shida, Giulia Punzi, Yasuaki Kobayashi, Takeaki Uno, and Hiroki Arimura. Finding diverse strings and longest common subsequences in a graph. In *35th Annual Symposium on Combinatorial Pattern Matching (CPM 2024)*, volume 296 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 27:1–27:19. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2024. doi: 10.4230/LIPIcs.CPM.2024.27. URL https://doi.org/10.4230/LIPIcs.CPM.2024.27.

C Spearman. The proof and measurement of association between two things. *The American Journal of Psychology*, 15 (1):72–101, 1904.

C Spearman. Footrule for measuring correlation. *British Journal of Psychology*, 2(1):89, 1906.

Dong Wei, Md Mouinul Islam, Baruch Schieber, and Senjuti Basu Roy. Rank aggregation with proportionate fairness. In *Proceedings of the 2022 International Conference on Management of Data*, SIGMOD '22, pp. 262–275, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450392495. doi: 10.1145/3514221.3517865. URL https://doi.org/10.1145/3514221.3517865.

Cai-Nicolas Ziegler, Sean M. McNee, Joseph A. Konstan, and Georg Lausen. Improving recommendation lists through topic diversification. In *Proceedings of the 14th International Conference on World Wide Web*, pp. 22–32. Association for Computing Machinery, 2005. doi: 10.1145/1060745.1060754. URL https://doi.org/10.1145/1060745.1060754.

Richard Zippel. Probabilistic algorithms for sparse polynomials. In *Proceedings of the International Symposiumon on Symbolic and Algebraic Computation*, EUROSAM '79, pp. 216–226, Berlin, Heidelberg, 1979. Springer-Verlag. ISBN 3540095195.

AI USE STATEMENT

We used LLMs as research, writing, and programming assistants during the development of this work. In particular, we used LLMs to formalize our ideas for the matching-based constructions underlying the diameter and min-dispersion algorithms, and to help develop and write the reductions used to establish NP-hardness. Starting from the observation that farthest-first traversal gives a factor-two approximation when supplied with an exact robust-farthest oracle, we developed an algebraic technique to construct such an oracle for fixed k; LLMs then helped us formalize its proof and develop and present the pairwise ex-ante guarantee for arbitrary k.

We also used LLMs to draft, edit, and improve the presentation of some formal arguments. We manually verified all AI-assisted proofs and claims, and rewrote most of the resulting material; we have also improved some of the reductions provided by the LLM. We additionally used LLM tools to assist with code optimization. We tested all AI-assisted code changes by comparing their outputs with those of the original implementations on the same test instances.

We take full responsibility for the final content of this work, including all text, proofs, claims, code, and other artifacts produced with generative AI assistance.

ETHICS STATEMENT

This work develops theoretical and algorithmic techniques intended to advance research in machine learning and rank aggregation. Although ranking systems may have broader societal consequences depending on their application and deployment, we do not identify any direct ethical concerns arising from the results of this paper that require specific discussion.

REPRODUCIBILITY STATEMENT

We provide proofs of all theoretical results in the appendix. The uploaded supplementary material contains the source code for all implemented algorithms, the exact dataset instances used in our experiments, and instructions for installing the required packages and reproducing the experimental results. The code uses the same random seed to ensure the reported plots and numerical results are reproducible.

A Other Related Work

For Kendall–tau, the classical Kemeny aggregation problem is NP-hard (Bartholdi et al., 1989; Dwork et al., 2001a); nevertheless, constant-factor approximation algorithms (Ailon et al., 2008) and a polynomial-time approximation scheme (Kenyon-Mathieu & Schudy, 2007) are known. This contrasts with the exact assignment algorithm for the Spearman-footrule median (Dwork et al., 2001a). Fair rank aggregation has been studied through group-representation constraints (Kuhlman & Rundensteiner, 2020; Chakraborty et al., 2022; Wei et al., 2022); recent work gives an exact algorithm for the fair top-k problem and a 2-approximation for the full footrule ranking problem (Chakraborty et al., 2026b), alongside approximation results for other ranking distances (Chakraborty et al., 2025).

Diversity maximization itself has a long history through max-sum and max-min dispersion in metric spaces (Ravi et al., 1994; Hassin et al., 1997; Cevallos et al., 2016). More recent work studies generic frameworks for generating diverse combinatorial solutions (Baste et al., 2022; Hanaka et al., 2023), as well as diverse matchings and structures in graphs and matroids (Fomin et al., 2024b;a), diverse strings and subsequences (Shida et al., 2024), and diverse satisfying assignments (Austrin et al., 2025; Misra et al., 2024). Closest to our setting are parameterized algorithms for diversity in Kemeny rank aggregation under Kendall–tau distance (Arrighi et al., 2021). Recent work also proposes producing a set of partial consensus rankings with ties and evaluates a metaheuristic for that task empirically (Aledo et al., 2025). Related theory has been developed for diverse exact and near-optimal Hamming medians (Chakraborty et al., 2026a). Together, these works leave open the polynomial-time approximation landscape for diverse rank aggregation under the Spearman footrule, which is the setting studied here.

B Assignment Formulation for Rank Aggregation

We briefly describe the reduction from rank aggregation to minimum cost weighted bipartite matching, as this will be key to our results. For candidate i and position p, define the assignment cost $c_{i,p} = \sum_{\pi \in S} |p - \pi(i)|$.

Then construct a complete bipartite graph with vertex sets $\mathcal{C} = [d]$ and $\mathcal{P} = [d]$, and for every $i \in \ candidates$ and $p \in \mathcal{P}$, add an edge with weight $c_{i,p}$. Then, for any ranking σ, one can construct a perfect matching where the edge (i, p) is in the matching if $\sigma(i) = p$, and the matching cost equals the objective cost of σ as the aggregate rank. Similarly, given any perfect matching, one can construct a corresponding ranking.

Consider the linear program that models an instance of minimum-cost perfect matching.

$$\begin{aligned} \min \quad & \sum_{i \in \mathcal{C}} \sum_{p \in \mathcal{P}} c_{i,p} x_{i,p} & \\ \text{subject to} \quad & \sum_{p \in \mathcal{P}} x_{i,p} = 1 & \text{for every } i \in \mathcal{C}, \\ & \sum_{i \in \mathcal{C}} x_{i,p} = 1 & \text{for every } p \in \mathcal{P}, \\ & x_{i,p} \geq 0 & \text{for every } i, p. \end{aligned} \tag{1}$$

Its dual is

$$\begin{aligned} \max \quad & \sum_{i \in \mathcal{C}} u_i + \sum_{p \in \mathcal{P}} v_p & \\ \text{subject to} \quad & u_i + v_p \leq c_{i,p} & \text{for every } i, p. \end{aligned} \tag{2}$$

Fix any optimal dual solution (u, v). Define the tight-edge graph

$$G_0 = (\mathcal{C} \cup \mathcal{P}, E_0), \qquad E_0 = \{(i, p) : u_i + v_p = c_{i,p}\}.$$

Lemma B.1 (Exact characterization by tight edges)**.** *A ranking belongs to $\mathcal{O}$ if and only if its perfect matching is contained in E_0.*

Proof. For every edge, let

$$s_{i,p} = c_{i,p} - u_i - v_p \geq 0$$

be its dual slack. Let M be any perfect matching. Because every candidate and every position occurs exactly once in M, strong duality gives

$$\begin{aligned}\mathrm{cost}(M) - \mathrm{OPT} &= \sum_{(i,p)\in M} c_{i,p} - \left(\sum_{i\in\mathcal{C}} u_i + \sum_{p\in\mathcal{P}} v_p\right) \\ &= \sum_{(i,p)\in M} (c_{i,p} - u_i - v_p) \\ &= \sum_{(i,p)\in M} s_{i,p}.\end{aligned}$$

All terms in the final sum are nonnegative. Hence the gap is zero if and only if every edge of M has zero slack, which is equivalent to $M \subseteq E_0$. □

C Exact Algorithm for Maximum Diameter Aggregate Rankings

As a warm-up, consider the diversity problem with $k = 2$, also referred to as the *maximum diameter* problem. This special case makes the main combinatorial idea particularly transparent. In particular, we will prove the following.

Theorem C.1. *There exists an algorithm that, given as input a set of rankings S over d candidates, finds two optimal aggregate rankings that maximize the distance between them. The algorithm runs in time $O(nd + d^3)$.*

Algorithm description. Consider the reduction of rank aggregation to minimum cost perfect matching on a bipartite graph G_0 with edge set E_0. Now, construct a bipartite graph H_2 as follows. Let its left vertex set contain two copies i^- and i^+ of every candidate $i \in [d]$, and its right vertex set contains two copies p^1 and p^2 of every position $p \in [d]$. For every tight edge $(i, p) \in E_0$, add the edges (i^-, p^1), (i^-, p^2), (i^+, p^1) and (i^+, p^2). Set the weight of the edges (i^-, p^1) and (i^-, p^2) to $-p$, and the weight of the edges (i^+, p^1) and (i^+, p^2) to p.

Next, find a maximum-weight perfect matching in H_2. Then partition the matching edges into two sets (see Lemma C.3 for details), such that each set describes a perfect matching between the candidate set and the position set. Finally, construct the corresponding rankings to the two matchings from the partitioning and return the two rankings as output. See Figure 3 for an illustration.

Algorithm 3 MaxDiameter(S)

1: Reduce rank aggregation on S to min-cost perfect matching on graph G.
2: Solve min-cost perfect matching on G, obtaining an optimal primal-dual solution.
3: Construct tight edge graph G_0 with edge set E_0.
4: Construct bipartite graph H_2.
5: $N^\star \leftarrow$ maximum-weight perfect matching on graph H_2.
6: Decompose $N^\star$ into perfect matchings M_1, M_2 from candidates to ranks.
7: $\sigma_1 \leftarrow$ ranking corresponding to M_1.
8: $\sigma_2 \leftarrow$ ranking corresponding to M_2.
9: **return** $\{\sigma_1, \sigma_2\}$.

Correctness analysis We first provide the arguments necessary to prove the correctness of our algorithm.

Lemma C.2. *For any maximum-weight perfect matching of H_2, for each candidate i, if i^- is matched to rank a_i and i^+ is matched to rank b_i, then $a_i \le b_i$. The contribution of candidate i to the weight of this matching is $b_i - a_i$.*

Proof. Suppose i^- is matched to a vertex representing rank a_i and i^+ is matched to a vertex representing rank b_i, with $a_i > b_i$. Swap the matching so that i^- is matched to the vertex representing rank b_i, and i^+ is matched to the vertex representing rank a_i. The new matching is feasible because i^- and i^+ have identical neighborhoods by construction. The contribution of the edges to the matching cost before is $b_i - a_i$ and $a_i - b_i$, which is strictly larger. Therefore, a maximum weight perfect matching must match i^- to a smaller rank than i^+, and the cost contribution follows from the construction of the edge weights. □

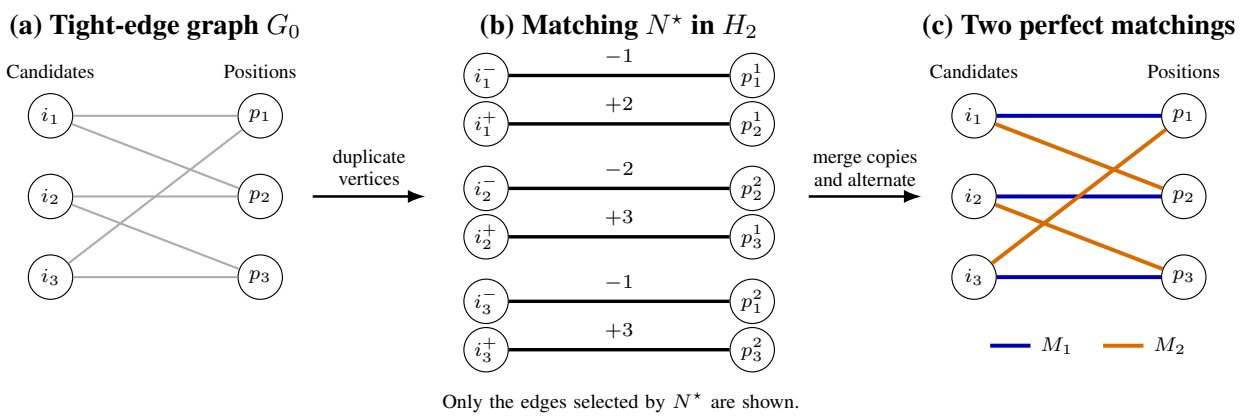


Figure 3: Illustration of the diameter construction for $d = 3$. Starting from the tight-edge graph G_0, we create two copies of every candidate and position. Edges incident to i^- and i^+ receive weights $-p$ and $+p$, respectively. Panel (b) shows only the edges selected by a maximum-weight perfect matching $N^\star$. Merging corresponding copies produces a 2-regular bipartite multigraph, whose alternating-cycle decomposition yields two perfect matchings M_1 and M_2, and hence two optimal aggregate rankings.

Lemma C.3. *Consider any perfect matching N of H_2. Then there exists an algorithm to partition N into perfect matchings M_1, M_2 from candidates to ranks, such that M_1 and M_2 preserve edges from copies to candidates to copies of positions.*

Proof. Construct a bipartite multigraph by merging i^- and i^+ into a single vertex i, and similarly merging p^1 and p^2 into a single vertex p, keeping the edges in N. After merging the vertices, every candidate is incident to the two edges selected for i^- and i^+ in the matching, and every position is incident to the two edges selected for p^1 and p^2. Thus every vertex has degree 2.

Every connected component of a 2-regular bipartite multigraph is an even cycle, where two parallel edges form a cycle of length two. Color the edges of every cycle alternately red and blue. Let the red edges be one perfect matching from candidates to ranks, and the blue edges be the second perfect matching. Clearly, at every vertex exactly one incident edge is red, and exactly one is blue. Thus, the red and blue edges form perfect matchings from candidates to ranks. By construction, all the edges belong to E_0, so the two matchings are perfect matchings of G_0. □

Running time analysis Here, we present the running-time argument for our algorithm.

Lemma C.4. *Algorithm 3 has running time $O(nd + d^3)$.*

Proof. The reduction from rank aggregation to minimum-cost perfect matching takes $O(nd + d^2)$ time. First, count how many ranks place candidate i in each position, taking $O(nd)$ time. Then, for any candidate i, the cost $c_{i,1}$ can be computed in $O(d)$ time, while the remaining costs $c_{i,j}$ for $j \geq 2$ can each be computed in $O(1)$ time, as $c_{i,p+1} - c_{i,p} = 2|\{\pi : \pi(i) \leq p\}| - n$. To see why, observe that the assignment cost of candidate i to rank $p + 1$ compared to rank p adds a cost of 1 for each ranking that placed i at rank p or before, and reduces the cost by 1 for all other rankings.

The Hungarian algorithm finds an optimal primal matching and optimal dual variables in $O(d^3)$ time. Therefore, running it on the constructed bipartite matching instance takes $O(d^3)$ time. The graph G_0 can be constructed in $O(d^2)$ time. The graph H_2 has $2d$ vertices; therefore, solving maximum cost perfect matching on it takes $O(d^3)$ time. Decomposing the perfect matching to two rankings takes linear time in the size of H_2. Thus, the algorithm runs in $O(nd + d^3)$ time. □

Now, we can prove Theorem C.1.

Proof of Theorem C.1. Let $N^\star$ be a maximum-weight perfect matching of H_2. By Lemma C.3, perfect matchings M_σ and M_τ between candidates to ranks can be obtained from $N^\star$. By Lemma B.1, their corresponding rankings σ and τ are optimal aggregate rankings.

Suppose that for any candidate i, i^- is matched to a_i and i^+ is matched to b_i in $N^\star$. Then in the output of our algorithm, i is placed at rank a_i in one of σ or τ, and placed at rank b_i in the other. Therefore, we have the following Spearman footrule distance between the two rankings.

$$\begin{aligned} F(\sigma,\tau) &= \sum_{i\in\mathcal{C}} |\sigma(i)-\tau(i)| \\ &= \sum_{i\in\mathcal{C}} |b_i - a_i| \\ &= \sum_{i\in\mathcal{C}} (b_i - a_i) && \text{(By Lemma C.2)} \\ &= w(N^\star). \end{aligned}$$

It remains to prove that no pair of optimal rankings has a larger distance. For the sake of contradiction, let $\sigma', \tau' \in \mathcal{O}$, where $F(\sigma',\tau') > F(\sigma,\tau)$. By Lemma B.1, all edges used by their matchings lie in E_0. Construct a perfect matching on H_2 in the following way. For each candidate i, assign the smaller of $\sigma'(i)$ and $\tau'(i)$ to i^-, and the larger to i^+. This forms a perfect matching N' of H_2, with $w(N') = F(\sigma',\tau') > F(\sigma,\tau) = w(N^\star)$, contradicting the optimality of $N^\star$.

By Lemma C.4, the algorithm runs in time $O(nd + d^3)$. □

D Correctness and Running Time of the Max-Sum Diversity Algorithm

Correctness analysis We first need a technical lemma about the edge weights.

Lemma D.1. *For integers $q_1 \le q_2 \le \cdots \le q_k$,*

$$\sum_{1\le a<b\le k} (q_b - q_a) = \sum_{t=1}^{k} \lambda_t q_t. \tag{3}$$

Proof. For any t, the term q_t appears positively once for each of the $t-1$ smaller indices and negatively once for each of the $k-t$ larger indices. Its total coefficient is therefore $(t-1)-(k-t) = 2t-k-1 = \lambda_t$. □

Next, we show that some maximum-weight perfect matching of H_k assigns the copies of every candidate to ranks in ascending order.

Lemma D.2. *There exists a maximum-weight perfect matching of H_k with the following property. If the copies of candidate i are matched through the edges $(i^1,p_1),(i^2,p_2),\ldots,(i^k,p_k)$, then $p_1 \le p_2 \le \cdots \le p_k$.*

Proof. Suppose a maximum-weight perfect matching $N^\star$ contains (i^s,p_s) and (i^t,p_t) for some $s<t$ and $p_s > p_t$. Replace these edges by (i^t,p_s) and (i^s,p_t). The result remains a perfect matching because all copies of candidate i have the same neighborhood. Its weight increases by $(\lambda_t-\lambda_s)(p_s-p_t) > 0$, since $p_s > p_t$ and $\lambda_t > \lambda_s$, contradicting the optimality of $N^\star$. □

We now show how to decompose a perfect matching of H_k into k perfect matchings between the original candidates and ranks.

Lemma D.3. *Consider a perfect matching N of H_k. There is an algorithm that finds k perfect matchings $M_1, M_2, \ldots, M_k$ from candidates to ranks, each using only edges in E_0, such that every edge $(i^t, p_{t'}) \in N$ projects to an edge contained in one of $M_1, M_2, \ldots, M_k$. The algorithm runs in time $O(kd^{5/2})$.*

Proof. Merge all copies of each candidate into a single vertex and all copies of each rank into a single vertex. The resulting graph H' is a k-regular bipartite multigraph. Therefore, for any subset of candidates in this graph, the set of neighbors must be at least as large. Therefore, applying Hall's theorem guarantees a perfect matching in the underlying simple graph. Find such a matching M_i, such as by applying the standard Hopcroft-Karp algorithm. Remove one copy of every edge in M_i from H', and repeat. After each removal, the remaining multigraph must still be regular and bipartite, so the process can be repeated k times successfully, and produces k perfect matchings whose union is precisely the projected edge multiset of N.

The algorithm invokes Hopcroft-Karp k times on a graph with $2d$ vertices and at most d^2 edges, for total time $O(kd^{5/2})$. □

Running time analysis Here, we prove the running time of Algorithm 1.

Lemma D.4 (k-tuple running time)**.** *Algorithm 1 has running time $O\left(nd + (kd)^3\right)$.*

Proof. As in Lemma C.4, the original reduction from rank aggregation to matching, an optimal dual solution, and E_0 can be obtained in $O(nd + d^3)$ time. The constructed graph H_k has $2kd$ vertices. Therefore, obtaining a maximum weight perfect matching $N^\star$ on H_k using an assignment algorithm takes time $O\left((kd)^3\right)$. From Lemma D.3, obtaining the k perfect matchings from $N^\star$ takes time $O(kd^{5/2})$. Finally, constructing the output rankings from the perfect matchings from candidates to ranks takes time $O(kd)$.

Taking the sum of the running times, it is clear our algorithm runs in time $O\left(nd + (kd)^3\right)$. □

We can now establish the correctness of Algorithm 1.

Proof of Theorem 3.1. Consider the output set $T = (\sigma_1, \sigma_2, \ldots, \sigma_k)$ of Algorithm 1. We have

$$\texttt{Obj}_{sum}(T) = \sum_{a<b} F(\sigma_a, \sigma_b) = \sum_{i\in[d]} \sum_{a<b} |\sigma_a(i) - \sigma_b(i)|.$$

Take any candidate i and sort the values $\sigma_1(i), \sigma_2(i), \ldots, \sigma_k(i)$ as $q_1 \le q_2 \le \cdots \le q_k$. Lemma D.1 gives

$$\sum_{a<b} |\sigma_a(i) - \sigma_b(i)| = \sum_{1\le a<b\le k} (q_b - q_a) = \sum_{t=1}^{k} \lambda_t q_t.$$

By Lemmas D.3 and D.2, the maximum cost perfect matching $N^\star$ on graph H_k constructed by the algorithm contains the edges $(i^1, q_1), (i^2, q_2), \ldots, (i^k, q_k)$. Hence $\texttt{Obj}_{sum}(T) = w(N^\star)$.

Suppose, for a contradiction, that some $T' \subseteq \mathcal{O}$ satisfies $\texttt{Obj}_{sum}(T') > \texttt{Obj}_{sum}(T)$. Every ranking in T' uses only edges of E_0 by Lemma B.1. For every candidate i, sort its ranks in the rankings of T' as $q_{i,1} \le \cdots \le q_{i,k}$ and assign i^t to $q_{i,t}$. These assignments form a perfect matching N' of H_k. By Lemma D.1,

$$\begin{aligned} w(N') = \sum_{i\in\mathcal{C}} \sum_{t=1}^{k} \lambda_t q_{i,t} &= \sum_{1\le a<b\le k} F(\sigma_a, \sigma_b) = \texttt{Obj}_{sum}(T') \\ &> \texttt{Obj}_{sum}(T) = w(N^\star), \end{aligned}$$

contradicting the optimality of $N^\star$. Thus T maximizes sum dispersion. Lemma D.4 gives the claimed running time. □

Remark D.5 (Encoding of k). An explicit output contains k rankings and hence has size $\Omega(kd)$. Thus a running time polynomial in the numerical value of k, rather than only in $\log k$, is unavoidable for an explicit-output algorithm.

E Analysis of the Max-Min Approximation Algorithms

We now show that Algorithm 2 gives an approximation algorithm for max-min diverse rank aggregation.

First, for any set $T \subseteq \mathcal{O}$ where $|T| \le k$, the robust-farthest radius is within factor 2 of $\texttt{OPT}_{min}$.

Lemma E.1. *For every $T \subseteq \mathcal{O}$ with $|T| < k$, it holds that $R(T) \ge \frac{\texttt{OPT}_{min}}{2}$.*

Proof. Fix an optimal solution $S^\star \subseteq \mathcal{O}$ of size k. Suppose for the sake of contradiction that $R(T) < \texttt{OPT}_{min}/2$. Then every ranking in $S^\star$ is at distance strictly less than $\texttt{OPT}_{min}/2$ from some member of T. Since $|T| < k$, by the pigeonhole principle, two rankings in $S^\star$ must be at distance strictly less than $\texttt{OPT}_{min}/2$ to the same member of T. By the triangle inequality, these two rankings have Spearman footrule distance between them strictly less than $\texttt{OPT}_{min}$, contradicting the definition of $S^\star$.

Thus $R(T) \ge \texttt{OPT}_{min}/2$. □

Now we can prove that as long as we have a sufficiently good robust-farthest oracle, our algorithm finds a set which is a good approximation of the optimal solution. Formally,

Lemma E.2. *Suppose that we have a α-robust-farthest oracle. Then the output set T_k of Algorithm 2 satisfies $\texttt{Obj}_{min}(T) \ge (\alpha/2)\texttt{OPT}_{min}$.*

Proof. Consider any $i \geq 2$. For the set T_{i-1} in the execution of the algorithm, Lemma E.1 and the definition of the α-robust-farthest oracle mean that the new ranking σ_i output by the oracle is at distance at least $\alpha\texttt{OPT}_{min}/2$ from all rankings in the set T_{i-1}.

Consider any $\sigma_i, \sigma_j \in T_k$ where $i < j$. Then when σ_j was added to the set T_{j-1}, it must hold that $F(\sigma_i, \sigma_j) \geq \alpha\texttt{OPT}_{min}/2$. Since this holds for any pair of rankings in T_k, the lemma follows. □

E.1 An exact robust-farthest oracle for fixed k

In this section, we describe an exact (i.e., $\alpha = 1$) robust-farthest oracle by using algebraic methods.

Fix any set $T = \sigma_1, \ldots, \sigma_t \subseteq \mathcal{O}$. For every tight edge $e = (i, p)$ in E_0, give it t values that encode the contribution to the Spearman footrule distance of placing candidate i at rank p to each of the rankings $\sigma_1, \ldots, \sigma_t$.

$$w^j_{i,p} = |p - \sigma_j(i)|, \qquad j \in [t].$$

For a perfect matching M, corresponding to a ranking σ, associate with it the t-dimensional distance vector

$$W(M) = \left(\sum_{(i,p)\in M} w^1_{i,p}, \ldots, \sum_{(i,p)\in M} w^t_{i,p} \right) = \big(F(\sigma, \sigma_1), \ldots, F(\sigma, \sigma_t)\big). \tag{4}$$

Let $B_d := \left\lfloor \frac{d^2}{2} \right\rfloor$ be the maximum distance of any two rankings under the Spearman footrule distance. Then every coordinate of $W(M)$ lies in $\{0, \ldots, B_d\}$.

Introduce a variable $r_{i,p}$ for every tight edge $(i, p) \in E_0$ and variables $z_1, \ldots, z_t$, where z_i tracks the Spearman footrule distance to the ranking σ_i. Form the matrix $\boldsymbol{A}$ with entries as follows

$$\boldsymbol{A}_{i,p} = \begin{cases} r_{i,p} \prod_{j=1}^{t} z_j^{w^j_{i,p}}, & (i,p) \in E_0, \\ 0, & (i,p) \notin E_0. \end{cases}$$

Lemma E.3. *For $v_1, v_2, \ldots, v_t$ where each $v_i \in \{0, \ldots, B_d\}$, the coefficient of the term $z_1^{v_1} z_2^{v_2} \ldots z_t^{v_t}$ in $\det(A)$ is the zero polynomial in the variables $\{r_e : e \in E_0\}$ if and only if no perfect matching M of G_0 has $W(M) = (v_1, \ldots, v_t)$.*

Proof. Expanding the determinant of $\boldsymbol{A}$ gives

$$\det(\boldsymbol{A}) = \sum_{\sigma \in \mathcal{S}_d} \operatorname{sgn}(\sigma) \prod_{i=1}^{d} r_{i,\sigma(i)} \prod_{j=1}^{t} z_j^{w^j_{i,\sigma(i)}}$$

where $\operatorname{sgn}(\sigma) = \{-1, 1\}$ is the parity of the inversions of σ. Observe that the edge variables ensure that for any two distinct rankings σ, σ', it holds that they have different edge monomials, as $\prod_{i=1}^{d} r_{i,\sigma(i)} \neq \prod_{i=1}^{d} r_{i,\sigma'(i)}$, thus avoiding cancellations upon summation for different rankings with the same z monomials.

Suppose some perfect matching M of G_0 has the distance vector $W(M) = (v_1, \ldots, v_t)$. Consider the corresponding ranking σ. Then by the definition of $\det(A)$, it must contribute the monomial $z_1^{v_1} z_2^{v_2} \ldots z_t^{v_t}$, and thus the coefficient is nonzero. The converse is also true. If the coefficient of the monomial $z_1^{v_1} z_2^{v_2} \ldots z_t^{v_t}$ is nonzero, then take the permutation/ranking σ that gave the term. Then σ must use only edges in E_0, and by definition of the $w^j_{i,p}$ values, the degree v_i gives the Spearman footrule distance of σ to σ_i.

If the coefficient is zero, then it must be that no perfect matching M of G_0 exists that can attain $W(M) = (v_1, \ldots, v_t)$. Any ranking that uses an edge not in E_0 is guaranteed to result in the zero polynomial due to corresponding entries of $\boldsymbol{A}$ being set to 0.

□

We will substitute the edge variables with field elements to compute the determinant only over the z variables. Therefore, we also need the Schwartz-Zippel lemma, widely used in polynomial identity testing.

Lemma E.4 (Schwartz-Zippel Lemma (Schwartz, 1980; Zippel, 1979)). *Let $p(x_1, x_2, \ldots, x_n)$ be a non-zero polynomial of total degree $d \geq 0$ over a field $\mathbb{F}$. Let S be a finite subset of $\mathbb{F}$, and let $r_1, r_2, \ldots, r_n$ be selected uniformly and independently at random from S. Then:*

$$\Pr[p(r_1, r_2, \ldots, r_n) = 0] \leq \frac{d}{|S|}.$$

Lemma E.5 (Randomized exact robust-farthest query). *Let a list of rankings $T = \{\sigma_1, \ldots, \sigma_t\} \subseteq \mathcal{O}$ and a $0 < \zeta < 1$ be given. There exists a probabilistic 1-robust-farthest oracle which returns a ranking τ that satisfies $\min_{\sigma \in T} F(\tau, \sigma) \geq R(T)$. The oracle succeeds with probability at least $1 - \zeta$, and runs in time*

$$(B_d + 1)^{O(t)} \operatorname{poly}(d, \log(1/\zeta)) = d^{O(t)} \operatorname{poly}(\log(1/\zeta)). \tag{5}$$

The oracle has one-sided detection error.

Proof. We will work over a finite field $\mathbb{F}$ satisfying

$$|\mathbb{F}| > \max\left\{B_d + 1, \frac{d(B_d + 1)^t}{\zeta}\right\}. \tag{6}$$

Such a field can be constructed with bits polynomial in the input length. Sample field elements uniformly at random and substitute them in place of each of the $r_{i,p}$ variables. After this substitution, $\det(\boldsymbol{A})$ has individual degree at most B_d in each z_j. We then compute the determinant over the random field elements and z variables.

Consider any distance vector $(v_1, \ldots, v_t)$ attained by some ranking σ using only tight edges. By Lemma E.3, the coefficient polynomial is a sum over terms of the form $\prod_{i=1}^{d} r_{i,\sigma(i)}$, which is of total degree d in the edge variables. By the Schwartz-Zippel lemma (Lemma E.4), if the variables $r_{i,p}$ are evaluated independently and uniformly over a finite field $\mathbb{F}$, the probability that this polynomial evaluates to zero after substitution by random field elements of $\mathbb{F}$ is at most $d/|\mathbb{F}|$. Therefore, by a union bound over at most $(B_d + 1)^t$ possible distance vectors, all distance vectors that are achieved by perfect matchings of G_0 have nonzero coefficients with probability at least $1 - \zeta$, by the choice of the size of $\mathbb{F}$. Therefore, for the remainder of this proof, we assume that all distance vectors attained by some σ appear in the determinant.

Among all monomials $z_1^{v_1} z_2^{v_2} \ldots z_t^{v_t}$ in the computed determinant, find the one which maximizes $\min_{j \in [t]} v_j$. A matching that has distance vector $(v_1, \ldots, v_t)$ can be recovered by self-reduction using the *same* random values of the r_e's, via the following method.

Pick any candidate i. Then for each rank p, perform the following. Delete row i and column p from $\boldsymbol{A}$ to form a smaller matrix $\boldsymbol{A}'$, and set $v'_j = v_j - w^j_{i,p}$ for each $j \in [t]$. Compute the determinant of $\boldsymbol{A}'$ and check that the distance vector v' is attainable. If so, then place candidate i at rank p in the output ranking; and recurse with $\boldsymbol{A}'$ and v'. This recovery introduces no additional probability of failure and requires only polynomially many coefficient computations.

As each coordinate of an attainable weight vector can take at most $B_d + 1$ distinct values, we have that the number of distinct monomials in the determinant after substitution of $r_{i,p}$ variables with random field elements is upper bounded by $(B_d + 1)^t$. A determinant algorithm can compute the determinant in time polynomial in $(B_d + 1)^t$, d and $\log(1/\zeta)$. The self-reduction adds at most d^2 determinant computations. This gives the claimed running time.

By Lemma E.3, a distance vector not achieved by any perfect matching of G_0 has coefficient identical to 0 and never appears in the determinant. Therefore, the oracle always returns a ranking in $\mathcal{O}$. □

Now, we have all the tools to prove Theorem 4.2.

Proof of Theorem 4.2. By Lemma E.2, the output of the algorithm T satisfies $\texttt{Obj}_{min}(T) \geq \frac{\texttt{OPT}_{min}}{2}$ if the robust-farthest oracle is 1-robust. We use the algorithm of Lemma E.5 as the robust-farthest oracle, where the failure probability is set to $\zeta/(k-1)$. By a union bound across all $k-1$ queries, all queries are correct with probability at least $1 - \zeta$. Therefore, with probability at least $1 - \zeta$, our algorithm outputs a set T where $\texttt{Obj}_{min}(T) \geq \frac{\texttt{OPT}_{min}}{2}$ for the max-min diversity rank aggregation problem with input S.

The largest query has $t = k - 1$, and so the running time is $d^{O(k)} \operatorname{poly}(\log(1/\zeta))$ for the oracle. Simply reading the input rankings takes time $O(nd)$, thus giving the stated running time. □

The algorithm is Monte Carlo only in its approximation guarantee. By Lemma E.3, every ranking obtained via the oracle must use only tight edges of G_0, and hence every output ranking always belongs to $\mathcal{O}$.

E.2 A POLYNOMIAL TIME APPROXIMATION ALGORITHM FOR ARBITRARY k

In this section, we provide an approximation robust-farthest oracle, which gives a $(2+\epsilon)$-approximation (for any $\epsilon > 0$) in expectation for the max-min diverse rank aggregation problem when used as the oracle in Algorithm 2.

The oracle makes use of the multiplicative weight update method. Therefore, we will need to consider a variant of the problem where we are given a set $T \subset \mathcal{O}$ and corresponding weights, and the objective is to find a ranking in $\mathcal{O}$ that maximizes the weighted sum of distances to T.

Lemma E.6. *Let $S \subseteq \mathcal{S}_d$ be the input rank aggregation instance. For $T = \{\sigma_1, \sigma_2, \ldots, \sigma_t\} \subseteq \mathcal{O}$ and a probability vector $\gamma \in \mathbb{R}^t_{\geq 0}$ with $\sum_j \gamma_j = 1$, a ranking $\tau \in \mathcal{O}$ maximizing $\sum_{j=1}^t \gamma_j F(\tau, \sigma_j)$ can be found in polynomial time.*

Proof. Reduce the instance of rank aggregation to weighted bipartite matching, and solve the assignment problem to construct the tight edge graph G_0. For a tight edge (i, p), set its weight equal to $\sum_{j=1}^t \gamma_j |p - \sigma_j(i)|$. Solve maximum weight perfect matching on G_0 with these weights, and output the ranking $\tau^\star$ corresponding to this matching. Observe that the weight of the matching is equal to $\sum_{j=1}^t \gamma_j F(\tau^\star, \sigma_j)$.

By Lemma B.1, $\tau^\star \in \mathcal{O}$, and $\tau^\star$ must be optimal, as any other ranking in $\mathcal{O}$ with a larger objective value must have a corresponding perfect matching of weight greater than that of $\tau^\star$. □

Multiplicative Weight Updates Oracle. Let us now describe a robust-farthest oracle using a multiplicative weight update approach. Recall that $B_d = \left\lfloor \frac{d^2}{2} \right\rfloor$, the largest distance between any two rankings of $\mathcal{S}_d$ under the Spearman footrule distance. Let $T = (\sigma_1, \ldots, \sigma_t)$, take an accuracy parameter $0 < \eta \leq B_d$, and assume that $t \geq 2$. Define $Z := \left\lceil \frac{4B_d^2 \log t}{\eta^2} \right\rceil$ and $\rho := \frac{\eta}{2B_d}$. The algorithm first initializes the weight for all rankings to be equal. Then for $s = 1, 2, \ldots, Z$, it does the following: find $\tau^s \in \mathcal{O}$ that maximizes the sum of weighted distances to T. Then, adjust the weights for each ranking so that rankings farther from τ^s have their weights reduced more than rankings closer to τ^s. Finally, return one of the $\tau^1, \tau^2, \ldots, \tau^s$ uniformly at random.

Algorithm 4 formally describes the algorithm.

Algorithm 4 MWUORACLE(T, Z)

1: Initialize $\beta_j^1 \leftarrow 1$ for all $j \in [t]$.
2: **for** $s = 1 \ldots Z$ **do**
3: $\quad \gamma_j^s \leftarrow \beta_j^s / \sum_{q=1}^t \beta_q^s$ for each $j \in [t]$.
4: $\quad$ Find τ^s maximizing $\sum_{j \in [t]} \gamma_j^s F(\tau^s, \sigma_j)$ using Lemma E.6.
5: $\quad \beta_j^{s+1} \leftarrow \beta_j^s \left(1 - \rho \frac{F(\tau^s, \sigma_j)}{B_d}\right)$ for all $j \in [t]$.
6: $\tau^\star \leftarrow$ uniformly pick at random from $(\tau^1, \tau^2, \ldots, \tau^Z)$.
7: **return** $\tau^\star$.

Now, we will prove that Algorithm 4 is an additive error robust-farthest oracle.

Lemma E.7. *Consider a $S \subseteq \mathcal{S}_d$ and a $T \subseteq \mathcal{O}$ (of size t) as input to Algorithm 4, with $\tau^\star$ as output. Then it holds that*

$$\min_{j \in [t]} \mathbb{E}\left[F(\tau^\star, \sigma_j)\right] \geq R(T) - \eta$$

where the expectation is over the randomness of the selection of $\tau^\star$.

Proof of Lemma E.7. Recall that the robust-farthest radius of the set T is defined as $R(T) = \max_{\tau \in \mathcal{O}} \min_{\sigma_j \in T} F(\tau, \sigma_j)$. Let $\tau \in \mathcal{O}$ attain this value. Since $F(\tau, \sigma_i) \geq R(T)$ for all $\sigma_i \in T$, it holds that $\sum_{j=1}^t \gamma_j^s F(\tau, \sigma_j) \geq R(T)$. By optimality of τ^s,

$$\sum_{j=1}^t \gamma_j^s F(\tau^s, \sigma_j) \geq R(T). \tag{7}$$

Let $\Phi_s = \sum_{j\in[t]} \beta_j^s$. By the update step of Algorithm 4,

$$\begin{aligned}
\Phi_{s+1} &= \sum_{j\in[t]} \beta_j^s \left(1 - \frac{\rho}{B_d} F(\tau^s, \sigma_j)\right) \\
&= \Phi_s - \frac{\rho}{B_d} \sum_{j\in[t]} \beta_j^s F(\tau^s, \sigma_j) \\
&= \Phi_s \left(1 - \frac{\rho}{B_d} \sum_{j\in[t]} \gamma_j^s F(\tau^s, \sigma_j)\right) \\
&\le \Phi_s \left(1 - \frac{\rho}{B_d} R(T)\right) \\
&\le \Phi_s \exp\left(-\frac{\rho R(T)}{B_d}\right).
\end{aligned}$$

As $\beta_j^1 = 1$ for all $j \in [t]$, consequently

$$\Phi_{Z+1} \le t \exp\left(-\frac{\rho Z R(T)}{B_d}\right). \tag{8}$$

As $\rho \le 1/2$, it follows that $0 \le \rho F(\tau^s, \sigma_j)/B_d \le 1/2$. For any $j \in [t]$,

$$\begin{aligned}
\log \beta_j^{Z+1} &= \sum_{s=1}^{Z} \log\left(1 - \rho \frac{F(\tau^s, \sigma_j)}{B_d}\right) \\
&\ge -\rho(1+\rho) \sum_{s=1}^{Z} \frac{F(\tau^s, \sigma_j)}{B_d}.
\end{aligned}$$

Combining this inequality with equation 8 and $\beta_j^{Z+1} \le \Phi_{Z+1}$ yields

$$\frac{1}{1+\rho}\left(R(T) - \frac{B_d \log t}{Z\rho}\right) \le \frac{1}{Z}\sum_{s=1}^{Z} F(\tau^s, \sigma_j).$$

Our choices $\rho = \eta/(2B_d)$ and $Z = \lceil 4B_d^2 \log(t)/\eta^2 \rceil$ imply

$$\frac{B_d \log t}{Z\rho} \le \frac{\eta}{2}, \qquad \rho R(T) \le \rho B_d = \frac{\eta}{2}. \tag{9}$$

Therefore,

$$R(T) - \eta \le \frac{1}{Z}\sum_{s=1}^{Z} F(\tau^s, \sigma_j).$$

A uniformly random selection of the output ranking has exactly this expected distance. Since the bound holds for every $j \in [t]$, the lemma follows. □

Now we show that using the multiplicative weight update algorithm as the oracle yields our stated approximate max-min diverse rank aggregation result.

Theorem E.8. *Let $S \subseteq \mathcal{S}_d$ and k be given as input to the max-min diverse rank aggregation problem, and a rational $0 < \eta \le B_d$. Let $\sigma_1 \in \mathcal{O}$. For $\ell = 2, \ldots, k$, condition on the rankings $(\sigma_1, \sigma_2, \ldots, \sigma_{\ell-1})$ and obtain σ_ℓ from Algorithm 4 with $\sigma_1, \ldots, \sigma_{\ell-1}$ as input. Then*

$$\min_{1\le a<b\le k} \mathbb{E}\left[F(\sigma_a, \sigma_b)\right] \ge \frac{OPT_{min}}{2} - \eta. \tag{10}$$

The algorithm has running time

$$O\left(nd + d^3 + \frac{(d^7 + kd^6)k \log k}{\eta^2}\right).$$

Proof of Theorem E.8. Fix $a < b$ and condition on the complete history of selected rankings immediately before σ_b is selected. Lemma E.1 gives that the robust-farthest value of the selected rankings is lower bounded by $R((\sigma_1, \ldots, \sigma_{b-1})) \geq \texttt{OPT}_{min}/2$, and Lemma E.7 therefore gives

$$\mathbb{E}\big[F(\sigma_b, \sigma_a) \mid (\sigma_1, \ldots, \sigma_{b-1})\big] \geq \frac{\texttt{OPT}_{min}}{2} - \eta.$$

Taking this expectation over the history of selected rankings proves equation 10.

Now let us consider the running time of the algorithm.

Constructing the bipartite matching instance in Lemma E.6 with a set T of size ℓ takes $O(\ell d^2)$ time, and solving it with a standard assignment algorithm takes $O(d^3)$ time. Therefore, each iteration of the for loop in Algorithm 4 requires $O(kd^2 + d^3)$ time. Further, we have that $Z = O\left(\frac{d^4 \log k}{\eta^2}\right)$. Therefore the running time of Algorithm 4 is $O\left(\frac{(kd^6+d^7)\log k}{\eta^2}\right)$. We run Algorithm 4 for $t = 2, \ldots, k$, and the initial preprocessing of finding the tight matching graph G_0 requires time $O(d^3 + nd)$. Therefore, we get $O\left(nd + d^3 + \frac{(kd^6+d^7)k \log k}{\eta^2}\right)$ for the running time.

For rationally encoded η, the multiplicative weights and scaled assignment profits have polynomial bit length: every round adds only polynomially many bits and the number of rounds is polynomial in $1/\eta$. Hence the corresponding bit complexity is polynomial as well. □

With this, we can prove our main theorem of a multiplicative approximation for max-min diverse rank aggregation.

Proof of Theorem 4.3. Apply Theorem E.8 with the parameter $\eta = 2\epsilon$, and let T be the output of the algorithm.

There are two cases. The first is that $\texttt{OPT}_{min} = 0$, so any solution set consisting of rankings from $\mathcal{O}$ is optimal.

Otherwise, it holds that $\texttt{OPT}_{min} \geq 2$, as any two distinct rankings have at least this distance. Therefore, $\texttt{OPT}_{min}\epsilon \geq 2\epsilon$, and it follows that

$$\min_{1 \leq a < b \leq k} \mathbb{E}[F(\sigma_a, \sigma_b)] \geq \frac{\texttt{OPT}_{min}}{2} - 2\epsilon \geq \left(\frac{1}{2} - \epsilon\right) \texttt{OPT}_{min}.$$

□

F Hardness of the Exact Robust-Farthest Query

We will reduce from the binary Closest String Problem, which is known to be NP-complete (Frances & Litman, 1997).

Definition F.1 (Closest String Problem)**.** Consider binary strings $s^1, s^2, \ldots, s^t \in \{0,1\}^h$ and an integer q. Does there exist a binary string $s^\star \in \{0,1\}^h$ such that $H(s^i, s^\star) \leq q$ for all $i \in [t]$, where $H(\cdot)$ represents the Hamming distance?

Consider an instance of the binary Closest String problem, with input strings $s^1, \ldots, s^t \in \{0,1\}^h$ and integer q.

Reduction. First, we construct two rankings over $2h$ candidates. Let the candidates be labeled $a_1, a_2, \ldots, a_h, b_1, b_2, \ldots, b_h$. The set S consists of the ranking $(a_1, b_1, a_2, b_2, \ldots, a_h, b_h)$, and the ranking $(b_1, a_1, b_2, a_2, \ldots, b_h, a_h)$.

It is easy to verify that any ranking which places a_i and b_i at ranks $2i-1$ and $2i$ (in either order) is an optimal aggregate ranking of S, as ranks $2i-1$ and $2i$ are the median ranks of a_i and b_i in the input S.

Now, we construct rankings from the strings by using candidates a_i and b_i to encode the i-th bit of the string. We encode bit 0 by placing a_i at rank $2i-1$, and b_i at rank $2i$. We encode bit 1 by placing b_i at rank $2i-1$ and a_i at rank $2i$. For example, if $s^1 = 1011$, then we construct a $\sigma_1 = (b_1, a_1, a_2, b_2, b_3, a_3, b_4, a_4)$. Clearly, each constructed $\sigma^i \in \mathcal{O}$. Further, an important observation is that the Spearman footrule distance between two such rankings is twice the Hamming distance between the strings, that is,

$$2H(s^i, s^j) = F(\sigma_i, \sigma_j). \tag{11}$$

This follows from the fact that, in our encoding, for each bit where the two strings s^i, s^j differ, the corresponding rankings σ_i, σ_j differ in distance by 1 for a_i and by 1 for b_i.

Then, the decision instance of robust-farthest query constitutes S, the rankings $(\sigma_1, \ldots, \sigma_t)$ and the value $D = 2(h-q)$.

Proof of Theorem 4.5. Note that the membership in NP is straightforward. Given S, one can compute the objective value of any optimal aggregate ranking and verify that a given certificate ranking σ' indeed attains this objective value. Then, verifying that $F(\sigma_i, \sigma') \geq D$ for each $\sigma_i \in T$ is straightforward.

Now, we argue the correctness of the reduction. First, we show that a yes instance of closest string maps to a yes instance of robust-farthest query.

Suppose there exists some string $s \in \{0,1\}^h$ that satisfies the closest string problem. Let $\bar{s}$ denote the complement of the string (i.e., flip each bit), and $\bar{\sigma}$ be the ranking obtained by performing the encoding as described before. Clearly, $\bar{\sigma}$ must be in $\mathcal{O}$. Consider the distance of $\bar{\sigma}$ from any σ_i. As $H(s, s^i) \leq q$, it must hold that $H(\bar{s}, s^i) \geq h - q$. From equation 11, we have that $F(\bar{\sigma}, \sigma_i) \geq 2(h-q)$. Since this holds for all $i \in [t]$, $\bar{\sigma}$ is a satisfying ranking for the constructed instance of robust-farthest query.

Now, we show that a yes instance of robust-farthest query maps to a YES instance of closest string. Suppose there exists some ranking $\bar{\sigma}$ such that $F(\bar{\sigma}, \sigma_i) \geq 2(h-q)$ for all $i \in [t]$. Then, by an argument similar to the above, we can construct a string $\bar{s}$ from our encoding. Let s be the complement of $\bar{s}$; as before, we can argue that $H(s, s^i) \leq q$ for all $i \in [t]$.

Finally, since S consists of two rankings of length $2h$ and each ranking σ_i can be constructed in time linear in h, this reduction runs in polynomial time. □

It is worth clarifying that this hardness result does not conflict with our results in Appendix E.1. There, our exact robust oracle required time exponential in t, but T can be encoded using $td \log d$ bits, and encoding a constant-sized S can be linear in d, for d candidates. Therefore, our result for the exact oracle requires time exponential in the input encoding. However, if t is a fixed constant, the running time is polynomial in the input length.

G Hardness of Max-Min Diverse Rank Aggregation

The proof has three steps: we start with a known NP-complete problem of extending an edge precoloring, then we reduce it to finding diverse weighted perfect matchings on a bipartite graph. Then, we show how, given a weighted bipartite graph G, we can construct an unweighted bipartite graph H with vertex set candidates and ranks, where perfect matchings of G are in bijection to perfect matchings of H; and further, the Spearman footrule distance between two perfect matchings (essentially rankings) of H encodes the weight of edges in their symmetric difference in G. Finally, we show that we can construct a set of rankings where a ranking is in $\mathcal{O}$ if and only if it uses only edges in H.

We start with the problem of extending a partial edge k-coloring on 3-regular bipartite graphs. An edge k-coloring of a graph colors edges with at most k colors such that any two edges sharing a vertex have different colors. The following problem is NP-complete (Fiala, 2003).

Definition G.1 (Extending 3-Edge Coloring for 3-regular Bipartite Graph)**.** Let $G = (U \cup V, E)$ be a 3-regular bipartite graph, $k = 3$, and a given precoloring col : $E \to \{1, 2, 3, \phi\}$ which is a (valid) partial 3-coloring on the edges, with uncolored edges mapped to ϕ. Does there exist a valid coloring of all edges which extends the given precoloring?

G.1 From Edge Precoloring to Weighted Matchings

We first use the fact that extending a partial proper edge coloring with three colors is NP-complete on 3-regular bipartite graphs (Fiala, 2003). A proper edge coloring assigns different colors to edges sharing a vertex.

We first make a small extension to show that completing a partial edge coloring is NP-hard on a k-regular bipartite graph with k colors.

Lemma G.2. *Fix any $k \geq 3$. Given a k-regular bipartite graph, k colors, and a partial edge coloring, extending it to a full edge coloring is NP-hard.*

Proof. Start with an instance of extending edge coloring for a 3-regular bipartite graph $G = (U \cup V, E)$. Add disjoint copies of $K_{3,3}$ with no precolored edges until each side has $h \geq k$ vertices, and denote this graph by G'. Now take the bipartite complement of G'; that is, a vertex $a \in U$ and $b \in V$ have an edge in the complement if and only if they have

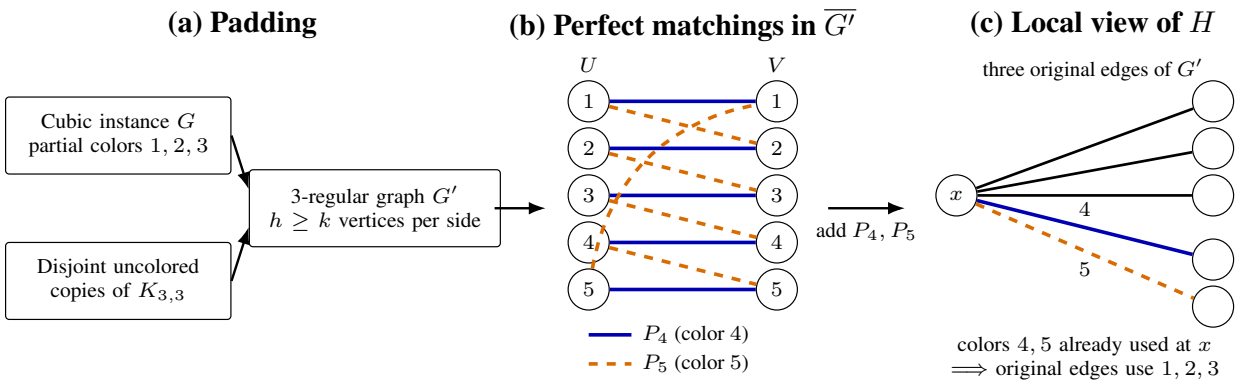


Figure 4: Reduction illustrated for $k = 5$. Padding gives a 3-regular graph G' with at least k vertices per side. We select $k - 3$ edge-disjoint perfect matchings from its $(h - 3)$-regular bipartite complement and precolor them with colors $4, \ldots, k$. These colors are then occupied at every vertex of H, forcing the original edges to use colors $1, 2, 3$.

no edge in G'. This complement graph is bipartite and must be $(h - 3)$-regular. Therefore, we can find $k - 3$ disjoint perfect matchings $P_4, \ldots, P_k$ in this complement graph. For each P_i, add its edges to G' and precolor every edge of P_i with color i. Denote the resulting graph H. Figure 4 illustrates the construction for $k = 5$; the same construction adds $k - 3$ disjoint perfect matchings for arbitrary fixed $k \geq 3$.

At every vertex, the added $k - 3$ edges already occupy colors $4, \ldots, k$, so any valid extension of the partial coloring of H must color the original edges of G using only colors $1, 2, 3$. Thus, it is a valid edge coloring of G that extends the input partial coloring.

Conversely, suppose the input coloring extends to a proper 3-edge-coloring of G. Then it extends to a proper 3-edge-coloring of G' (the subgraphs $K_{3,3}$ can be easily edge colored with 3 colors) and adding the precolored matchings $P_4, \ldots, P_k$ clearly yields a proper k-edge-coloring of H extending its precoloring. So the reduction is correct. The construction clearly runs in polynomial time. □

Now, we reduce the problem of completing a partial k-edge coloring on a k-regular bipartite graph to a diverse weighted perfect matching problem on bipartite graphs.

Definition G.3 (k-Diverse Weighted Perfect Matching on Bipartite Graphs)**.** Let $G = (U \cup V, E)$ be a weighted bipartite graph with integer weights w_e for each edge e, and let a threshold T be given as input. Does there exist k perfect matchings $M_1, \ldots, M_k \subseteq E$ such that for any M_i, M_j it holds that $\sum_{e \in (M_i \triangle M_j)} w_e \geq T$?

For brevity, denote $\sum_{e \in E'} w_e$ as $w(E')$ for any subset of edges E'.

Lemma G.4. *Take any constant $k \geq 3$. The problem of k-diverse weighted perfect matching on bipartite graphs is NP-hard. This holds even when G is k-regular and all edge weights and T are positive integers polynomially bounded in the number of vertices, with the polynomial allowed to depend on k.*

Proof. Take any instance of extending a partial edge k-coloring on a k-regular bipartite graph G. Let r_j be the number of edges precolored j, and define the following values

$$p = \sum_{j=1}^{k} p_j, \qquad r = p + 1, \qquad d_j = p_j r^{j-1}, \qquad B = \sum_{j=1}^{k} d_j.$$

Form a graph H by adding a disjoint copy of $K_{k,k}$ to G and fix an ordering of the vertices. Choose the first vertex on the left of this new component and label its incident edges such that its edge to the first vertex on the right is f_1, the edge to the second vertex is f_2, and so on.

Let n be the number of vertices on each side of H. Set $A = kB$, and the diversity threshold $T = 2nA + 2B$.

Now, we give the edges the following weights.

$$w_e = \begin{cases} A & e \text{ is not precolored in } G, \\ A + r^{j-1} & e \text{ is precolored } j \text{ in } G, \\ A + B - d_j & e \text{ is the special edge } f_j. \end{cases}$$

Essentially, each receives a base weight A, ensuring that any two matchings whose symmetric difference in weight exceeds the diversity threshold are disjoint. We give additional weights to edges that are precolored and to the special edges f_i. We choose the weight so that the total additional weight equals exactly B for each color.

The special edges $f_1, \ldots, f_k$ are a gadget to ensure that for any color j, taking the corresponding edge f_j along with precolored edges gives exactly total additional weight B - otherwise, different colors will contribute different amounts of additional weight from precolored edges.

Suppose that we have a yes instance of extending edge coloring on G. Observe that as the graph G is k-regular, taking all edges of color $c \in [k]$ results in a perfect matching in G, as for any vertex, exactly one of its incident edges is color c. Now, consider the additional $K_{k,k}$. It has no precolored edges, and so we can color it such that f_j receives color j. Then color the remaining edges to obtain a valid edge coloring; this can be done simply by adding an offset of i to the cycle $1 \ldots k$ for the vertex $i + 1$ on the left and using these as the color to vertices in order on the right partition.

It is clear that then taking all edges of any color on H gives a perfect matching. For any color $c \in [k]$, let the set of edges colored c be M_c. Take the matchings $M_1, \ldots, M_k$ as the k diverse perfect matchings. Observe that for any M_i, the weight of its edges is

$$\begin{aligned} w(M_i) &= (n - p_i - 1)A + p_i \cdot (A + r^{i-1}) + A + B - p_i r^{i-1} \\ &= nA + B. \end{aligned}$$

Take any M_a, M_b. As they are edge-disjoint, the value $w(M_a \triangle M_b)$ is simply the sum of their weights, and so it holds that $w(M_a \triangle M_b) = 2nA + 2B \geq T$.

Conversely, suppose that we have a yes instance of k-diverse weighted perfect matchings. Let $M_1, \ldots, M_k$ be the perfect matchings that satisfy the instance. Clearly, each matching must contain exactly n edges. Suppose for the sake of contradiction that there exists at least one pair that shares an edge. Fix this pair to be M_i, M_j. Since they share at least one edge, $|M_i \triangle M_j| \leq 2n - 2$, and the base weights of A contribute at most $(2n - 2)A$ to $w(M_i \triangle M_j)$. Now, the amount of total additional weight placed on all edges is kB. Therefore, the amount of weight on the symmetric difference can be bounded as follows

$$\begin{aligned} w(M_a \triangle M_b) &\leq (2n - 2)A + kB \\ &< 2nA + 2B = T, \end{aligned}$$

where the strict inequality follows from $2A = 2kB$. Thus it must be that the matchings are edge disjoint. Now, let us inspect the additional weight of edges in each matching. For any matching M_a, let Q_a denote the additional weight, that is $Q_a = w(M_a) - nA$. Then we have that $Q_a + Q_b \geq 2B$, by the fact that $T = 2nA + 2B$, and the fact that the matchings form a satisfying solution. Further, $\sum_{a=1}^{k} Q_a = kB$ as that is the total additional weight across all the edges.

Now, we argue that $Q_a + Q_b = 2B$. Let us introduce a slack variable to the inequality earlier to obtain $Q_a + Q_b - s_{a,b} = 2B$.

Then we can write

$$\begin{aligned} \sum_{a=1}^{k} \sum_{b=a+1}^{k} s_{a,b} &= \sum_{a=1}^{k} \sum_{b=a+1}^{k} (Q_a + Q_b - 2B) \\ &= \sum_{a=1}^{k} \sum_{b=a+1}^{k} (Q_a + Q_b) - k(k-1)B && \text{(As } B \text{ is independent of the summation)} \\ &= (k-1) \sum_{a=1}^{k} Q_a - k(k-1)B && \text{(Since each } Q_a \text{ appears exactly } k-1 \text{ times)} \\ &= k(k-1)B - k(k-1)B \\ &= 0 \end{aligned}$$

Therefore, it must hold for every a, b that $Q_a + Q_b = 2B$. As $k \geq 3$, this implies $Q_a = B$ for every a. To see why, take any Q_a, Q_b, Q_c. Then

$$Q_a + Q_b = Q_a + Q_c = Q_b + Q_c = 2B$$

which implies $Q_a = Q_b = Q_c = B$.

Now, each matching contains exactly one of the special edges f_j. Let the matching containing f_j be M_j. Its additional weight from edges of G must be $d_j = p_j r^{j-1}$. Now, we argue that it can only contain precolored edges of color j. If it contains any precolored edge of color $\ell > j$, then it has additional weight $r^{\ell-1} > p_j r^{j-1}$, as r is strictly larger than the number of precolored edges. Suppose it contains some edges of color $\ell < j$. Then the total amount of additional weight from these edges can be at most $pr^{j-2} < r^{j-1}$. Therefore, M_j must contain every edge precolored j, and no edges precolored by any other color.

Now, for every $j \in [k]$, color the edges $M_j \cap E(G)$ with color j. It must be a valid edge coloring, as there are k such perfect matchings which are edge disjoint, and we have argued this coloring extends the given precoloring.

Let v be the number of vertices in G. Then, as it is k-regular, it has $kv/2$ edges and so $p \leq kv/2$. Further, k is fixed, and it holds that $B \leq p(p+1)^{k-1}$, so the weight of any edge is polynomially bounded by the length of the input. Therefore, every step is polynomial-time. In particular, this argument does not encode exponentially large weights in an instance with fixed k. □

G.2 Edge Weights to Spearman footrule Distance

The following construction allows us to move from diversity on the weights in the symmetric difference of matchings to Spearman footrule distance between two perfect matchings, or equivalently, rankings.

Lemma G.5. *Let $G = (U \cup V, E)$ be bipartite, with $|U| = |V| = d$, $|E| = m$, a perfect matching, and positive integer edge weights. One can construct a bipartite graph $H = ([\hat{d}] \cup [\hat{d}], \hat{E})$ (where $\hat{d}$ depends on G) whose perfect matchings are in bijection with those of G. There exists a positive integer C such that the following holds. For any two perfect matchings M, N on G, with bijection to perfect matchings M', N' respectively on H, and denoting $\sigma_{M'}, \sigma_{N'}$ as the corresponding rankings,*

$$F(\sigma_{M'}, \sigma_{N'}) = Cw(M \triangle N) + \epsilon(M, N), \qquad 0 \leq \epsilon(M, N) < C \tag{12}$$

Hence, for every integer T,

$$F(\sigma_{M'}, \sigma_{N'}) \geq CT \quad \iff \quad w(M \triangle N) \geq T. \tag{13}$$

The construction has size polynomial in d, m and the largest weight of any edge.

Proof. Let us index vertices of U as $u_1, \ldots, u_d$, vertices of V as $v_1, \ldots, v_d$, and edges of E as $e_1, \ldots, e_m$.

First, let

$$R = 2dm + 2d, \qquad C = 2dR.$$

For each edge $e_k = (u_i, v_j)$, we will associate with it two gadget candidates a_k, b_k, as well as two gadget ranks p_k, q_k. Note that these will all naturally be integers.

$$a_k = p_k = 2dk + 2d, \qquad b_k = q_k = \frac{Cw_{e_k} + p_k + 2j}{2} = \frac{C}{2} w_{e_k} + dk + d + j.$$

Further, let $\hat{d} = \max_{k \in [m]} q_k$.

Now, we will construct an unweighted bipartite graph $H = ([\hat{d}] \cup [\hat{d}], \hat{E})$. Therefore, matchings on it can be interpreted as rankings over the candidate set $[\hat{d}]$.

For each edge $e_k = (u_i, v_j)$ of G, we construct a path of five edges as follows.

$$2i \; -- \; p_k \; -- \; a_k \; -- \; q_k \; -- \; b_k \; -- \; 2j.$$

Note that essentially, vertex u_i maps to candidate $2i$, and vertex v_j maps to rank $2j$. See Figure 5 for an example.

Observe that as $2j \leq 2d$, we ensure that all vertices are mapped to a candidate and rank that are not used by any of the p_k or q_k. Further, clearly all p_k are distinct for each k and are at most R; while every q_k is larger than R, because

$w_e \geq 1$ and $C/2 > R$, so no p_k can collide with any q_k. For any $e_k = \{u_i, v_j\}, e_{k'} = \{u_{'i}, v_{j'}\}$ with the same weight, $dk + d + j$ and $dk' + d + j'$ are distinct as $1 \leq j \leq d$ so they must map to different q_k. If they have different weights, then since $dk + d + j < C/2$, we have $q_k \neq q'_k$.

Finally, all integers in $[\hat{d}]$ that are unused in the construction above will be dummy candidates and ranks; for each such dummy a, connect a single edge from a on the candidate vertices to a on the rank vertices. Therefore, any perfect matching must always take this edge, so we can safely ignore them in the remainder of this proof.

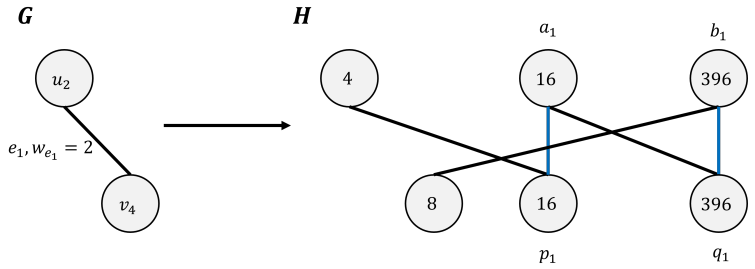


Figure 5: Example of the construction for an edge $e_1 = (u_2, v_4)$ with weight 2 where $d = 4, m = 5$, therefore $R = 48, C = 384$. Labels such as a_1 are solely for clarity; the vertices of H are identified only by the integers. Either the black edges or the blue edges must be taken in a perfect matching of H.

In a perfect matching of H, a path in our construction has exactly two possible states, which will correspond to whether $e_k = (u_i, v_j)$ is in the matching on G,

$$\begin{array}{r|l} e_k \text{ selected} & (2i, p_k),\ (a_k, q_k),\ (b_k, 2j) \\ e_k \text{ unselected} & (a_k, p_k),\ (b_k, q_k). \end{array}$$

Take any perfect matching of G, and follow this construction for each edge $e_k \in E$. Clearly, every candidate in the construction is matched to some rank, giving a unique perfect matching of H. Now consider the converse. For any perfect matching of H, if it contains the edges $(2i, p_k),\ (a_k, q_k),\ (b_k, 2j)$, then match u_i with v_j in G. Since every candidate vertex $2i$ must be matched to some rank p_k in a perfect matching on H, and every rank vertex $2j$ must be matched to some candidate b_k, we obtain a unique perfect matching on G. This proves the bijection.

Now, suppose we have two different perfect matchings M, N on G, with bijections to matchings M', N' on H. Consider any edge $e_k \in M \setminus N$.

Now, in the M' we have the matchings (a_k, q_k) and $(b_k, 2j)$, and in N' we have the matchings $(a_k, p_k), (b_k, q_k)$. Therefore, the contribution to the Spearman footrule distance by the change in ranks of the candidates a_k, b_k is equal to

$$|q_k - p_k| + |q_k - 2j| = 2q_k - p_k - 2j = Cw_{e_k}.$$

The case for $e_k \in N \setminus M$ is identical.

If an edge is in both M and N or if an edge is not in $M \cup N$, then a_k, b_k are assigned the same ranks in M' and N', and so the contribution to the distance is zero. Consider now the contribution to the Spearman footrule distance by candidate vertices $2i$ for $i \in [d]$. They can only be matched to some p_k, and we have that $p_k \leq R$. Therefore, the contribution is upper bounded by

$$\epsilon(M, N) \leq dR < C,$$

proving equation 12. Integrality of the weights on the edges then implies equation 13. The number of vertices of H is $2\hat{d} \leq C \max_{e_k \in E} w_{e_k} + R + 2d$, which is polynomial in w_{e_k}, d, m. □

G.3 Bipartite Perfect Matching to Rank Aggregation

We showed that a diverse weighted perfect matchings problem on bipartite graphs is NP-hard, and then argued that it is equivalent to considering perfect matchings on an unweighted graph that encode the Spearman footrule distance. We now show how to construct a set of rankings where a ranking is an optimal aggregate ranking only if it is a perfect matching on any arbitrary bipartite graph; this will allow us to complete the reduction to show max-min diverse rank aggregation is indeed NP-hard.

Lemma G.6. *Let $G = ([d] \times [d], E)$ be a bipartite graph. If G has a perfect matching, then one can construct, in time polynomial in d, a set S of rankings over the candidates $[d]$ such that a ranking $\sigma \in \mathcal{O}_S$ if and only if its corresponding matching M_σ is a valid perfect matching of G.*

Let us first define some useful notations. Let $\mathbf{1} \in \mathbb{R}^d$ be the all-ones vector, $\boldsymbol{J} = \mathbf{1}\mathbf{1}^\mathsf{T}$ be the all-ones matrix, and define the centering matrix

$$\boldsymbol{Q} = \boldsymbol{I} - \frac{1}{d}\boldsymbol{J}.$$

Given any $\boldsymbol{x}$, the vector $\boldsymbol{Q}\boldsymbol{x}$ is equal to $\boldsymbol{x}$ with the mean of its entries subtracted from every entry. In particular,

$$\boldsymbol{Q}\mathbf{1} = 0, \qquad \mathbf{1}^\mathsf{T}\boldsymbol{Q} = 0, \qquad \boldsymbol{Q}^2 = \boldsymbol{Q}.$$

Let Δ be the distance matrix of a path on d vertices, that is, $\Delta_{i,p} := |i - p|$, and let $\boldsymbol{L}_d$ be its Laplacian matrix. That is, $\boldsymbol{L}_d$ has entries of 1 at $(1, 1)$ and (d, d), 2 on the other diagonal entries, and -1 in all entries adjacent to the diagonal entries.

We first give a broad idea of the proof for this lemma.

Consider a set of rankings S. Define a matrix $\boldsymbol{A}$ where $\boldsymbol{A}_{a,r} := |\{\pi \in S : \pi(a) = r\}|$. That is, the entry contains the number of rankings that place candidate a at rank r. Consider the following matrix product $\boldsymbol{A}\Delta$. Observe that the entry $(\boldsymbol{A}\Delta)_{i,j}$ gives the contribution to the Spearman footrule cost by placing candidate i at rank j in an aggregate ranking, as entries of Δ capture the distance moved by candidates, while entries of $\boldsymbol{A}$ indicate the number of candidates at each rank. Therefore, solving minimum cost perfect matching with this as the weight matrix for a complete bipartite graph gives an optimal aggregate ranking of S.

Now, take some arbitrary bipartite graph $G = (U \cup P, E)$ (where $|U| = |P|$), and consider the zero-one matrix $\boldsymbol{W}$, where

$$W_{i,p} = \begin{cases} 0, & (i, p) \in E, \\ 1, & (i, p) \notin E. \end{cases}$$

Then, by using $\boldsymbol{W}_{i,j}$ as the weight for edge (i, j) in a minimum cost perfect matching instance, it is clear that a minimum cost perfect matching of cost 0 must be a matching that uses only edges of G. Our goal will be to construct a set of rankings S where the corresponding matrix $\boldsymbol{A}\Delta$ has exactly the same minimum cost perfect matchings as $\boldsymbol{W}$. Then any optimal aggregate ranking of S corresponds to perfect matchings on G.

To achieve this, the following lemma will be particularly helpful.

Lemma G.7. *For a $d \times d$ matrix $\boldsymbol{Z}$, the equality $\boldsymbol{Q}\boldsymbol{Z}\boldsymbol{Q} = 0$ holds if and only if there are vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^d$ such that*

$$\boldsymbol{Z}_{i,p} = \boldsymbol{a}_i + \boldsymbol{b}_p \qquad \forall i, p \in [d].$$

Proof. First suppose that $\boldsymbol{a}, \boldsymbol{b}$ exist such that $\boldsymbol{Z}_{i,p} = \boldsymbol{a}_i + \boldsymbol{b}_p$. Therefore, we can write $\boldsymbol{Z} = \boldsymbol{a}\mathbf{1}^\mathsf{T} + \mathbf{1}\boldsymbol{b}^\mathsf{T}$.

Using $\boldsymbol{Q}\mathbf{1} = 0$ and $\mathbf{1}^\mathsf{T}\boldsymbol{Q} = 0$, we obtain

$$\begin{aligned} \boldsymbol{Q}\boldsymbol{Z}\boldsymbol{Q} &= (\boldsymbol{Q}\boldsymbol{a})(\mathbf{1}^\mathsf{T}\boldsymbol{Q}) + (\boldsymbol{Q}\mathbf{1})(\boldsymbol{b}^\mathsf{T}\boldsymbol{Q}) \\ &= 0. \end{aligned}$$

Now, we show the converse.

$$r_i = \frac{1}{d}\sum_{p=1}^{d} \boldsymbol{Z}_{i,p}, \qquad s_p = \frac{1}{d}\sum_{i=1}^{d} \boldsymbol{Z}_{i,p}, \qquad \mu = \frac{1}{d^2}\sum_{i=1}^{d}\sum_{p=1}^{d} \boldsymbol{Z}_{i,p}$$

be, respectively, the mean of row i, the mean of column p, and the mean of all entries of $\boldsymbol{Z}$. By the definition of $\boldsymbol{Q}$, we will have that

$$(\boldsymbol{Q}\boldsymbol{Z}\boldsymbol{Q})_{i,p} = \boldsymbol{Z}_{i,p} - r_i - s_p + \mu.$$

Therefore, if $\boldsymbol{Q}\boldsymbol{Z}\boldsymbol{Q} = 0$, then

$$\boldsymbol{Z}_{i,p} = r_i + s_p - \mu.$$

Taking $\boldsymbol{a}_i = r_i$ and $\boldsymbol{b}_p = s_p - \mu$ proves the claim. □

To see the purpose of the preceding lemma, suppose we have two cost matrices $\boldsymbol{C}$ and $\boldsymbol{W}$ such that the following holds given some vectors $\boldsymbol{a}$, $\boldsymbol{b}$ and $\lambda > 0$

$$\boldsymbol{C}_{i,p} = \lambda \boldsymbol{W}_{i,p} + \boldsymbol{a}_i + \boldsymbol{b}_p.$$

For any perfect matching M from $[d]$ to $[d]$, its cost under $\boldsymbol{C}$ is

$$\sum_{i=1}^{d} \boldsymbol{C}_{i,M(i)} = \lambda \sum_{i=1}^{d} \boldsymbol{W}_{i,M(i)} + \sum_{i=1}^{d} \boldsymbol{a}_i + \sum_{i=1}^{d} \boldsymbol{b}_{M(i)}$$

The last two terms are independent of $M(i)$, and every perfect matching incurs exactly that cost. Thus $\boldsymbol{C}$ and $\boldsymbol{W}$ have exactly the same minimum cost perfect matchings. By Lemma G.7, verifying that $\boldsymbol{Q}(\boldsymbol{C} - \lambda \boldsymbol{W})\boldsymbol{Q} = 0$ allows us to conclude that $\boldsymbol{C}$ and $\boldsymbol{W}$ have the same minimum cost perfect matchings.

We also need a technical lemma that will simplify calculations later.

Lemma G.8. *The matrices $\boldsymbol{L}_d$, Δ, and $\boldsymbol{Q}$ satisfy*

$$\boldsymbol{L}_d \Delta \boldsymbol{Q} = -2\boldsymbol{Q}.$$

Proof. We first compute $\boldsymbol{L}_d \Delta$.

Observe that for any p,

$$(\boldsymbol{L}_d \Delta)_{r,p} = \begin{cases} |1-p| - |2-p|, & r = 1 \\ 2|r-p| - |r-1-p| - |r+1-p|, & 1 < r < d \\ |d-p| - |d-1-p|, & r = d. \end{cases}$$

We can observe that if $p = 1$, then $|1-p| - |2-p| = -1$, and is equal to 1 otherwise. Similarly, if $p = r$, the $2|r-p| - |r-1-p| - |r+1-p| = -2$, and is equal to 0 otherwise. Also, if $p = d$, then $|d-p| - |d-1-p| = -1$, and is 1 otherwise.

Consequently, we can write the product in the following form, where $\boldsymbol{e}_1$ and $\boldsymbol{e}_d$ are the elementary vectors,

$$\boldsymbol{L}_d \Delta = -2\boldsymbol{I} + (\boldsymbol{e}_1 + \boldsymbol{e}_d)\mathbf{1}^\mathsf{T}.$$

It holds that $(\boldsymbol{e}_1 + \boldsymbol{e}_d)\mathbf{1}^\mathsf{T}\boldsymbol{Q} = 0$ because $\mathbf{1}^\mathsf{T}\boldsymbol{Q} = 0$. Hence $\boldsymbol{L}_d \Delta \boldsymbol{Q} = -2\boldsymbol{Q}$, proving the lemma. □

We can now give the construction of S.

Proof of Lemma G.6. Construct a matrix $\boldsymbol{W}$ by setting its entries as follows.

$$\boldsymbol{W}_{i,p} = \begin{cases} 0, & (i,p) \in E, \\ 1, & (i,p) \notin E. \end{cases}$$

Any minimum cost perfect matching using $\boldsymbol{W}$ as the edge costs for a complete bipartite graph must be a perfect matching on the graph G.

As discussed, our goal is to construct a matrix $\boldsymbol{A}$ such that using $\boldsymbol{A}\Delta$ as the cost matrix for minimum-cost perfect matching yields all such perfect matchings as minimum-cost perfect matchings when using $\boldsymbol{W}$ as the cost matrix. Then if all entries of $\boldsymbol{A}$ are positive, and it can be written as the sum of permutation matrices, we can indeed construct a set of rankings S that gives $\boldsymbol{A}$.

We begin with the matrix $\boldsymbol{Y} = \boldsymbol{W}\boldsymbol{L}_d$. Because $\boldsymbol{L}_d$ is the Laplacian matrix of the path, multiplication by $\boldsymbol{L}_d$ takes a discrete second difference along every row of W. That is,

$$\boldsymbol{Y}_{i,p} := \begin{cases} \boldsymbol{W}_{i,1} - \boldsymbol{W}_{i,2}, & p = 1, \\ 2\boldsymbol{W}_{i,p} - \boldsymbol{W}_{i,p-1} - \boldsymbol{W}_{i,p+1}, & 1 < p < d, \\ \boldsymbol{W}_{i,d} - \boldsymbol{W}_{i,d-1}, & p = d. \end{cases}$$

Since every entry of $\boldsymbol{W}$ belongs to $\{0, 1\}$, it follows that

$$|Y_{i,p}| \leq 2 \qquad \text{for every } i, p.$$

Now define

$$X = -d\boldsymbol{Q}\boldsymbol{W}\boldsymbol{L}_d. \tag{14}$$

Equivalently, in terms of entries of $\boldsymbol{Y}$,

$$\boldsymbol{X}_{i,p} = -d\boldsymbol{Y}_{i,p} + \sum_{j=1}^{d} \boldsymbol{Y}_{j,p}.$$

Now, for a matrix where the entry (i, p) is the number of rankings of S that place candidate i at rank p, it must hold that the sum of every row and the sum of every column are identical. We show that $\boldsymbol{X}$ meets this requirement, in that the sum of each of its rows and columns must be 0.

First, every row of $\boldsymbol{X}$ sums to zero, as $\boldsymbol{Y}\mathbf{1} = 0$. To see why, we have that

$$\begin{aligned}\sum_{p=1}^{d} \boldsymbol{Y}_{i,p} &= \boldsymbol{W}_{i,1} - \boldsymbol{W}_{i,2} + \sum_{p=2}^{d-1}(2\boldsymbol{W}_{i,p} - \boldsymbol{W}_{i,p-1} - \boldsymbol{W}_{i,p+1}) - \boldsymbol{W}_{i,d-1} + \boldsymbol{W}_{i,d} \\ &= 0\end{aligned}$$

as all the terms cancel out.

Therefore,

$$\boldsymbol{X}\mathbf{1} = -d\boldsymbol{Q}\boldsymbol{W}\boldsymbol{L}_d\mathbf{1} = 0.$$

Every column of $\boldsymbol{X}$ also sums to zero, as using the fact that $\mathbf{1}^\mathsf{T}\boldsymbol{Q} = 0$,

$$\mathbf{1}^\mathsf{T}\boldsymbol{X} = -d\mathbf{1}^\mathsf{T}\boldsymbol{Q}\boldsymbol{W}\boldsymbol{L}_d = 0.$$

However, entries of $\boldsymbol{X}$ can be negative. To make all the entries at least 0, observe that as entries of $\boldsymbol{W}$ are either 0 or 1, it holds that $|\boldsymbol{Y}_{i,p}| \leq 2$ for any i, p. Therefore for any (i, p),

$$\begin{aligned}\boldsymbol{X}_{i,p} &= -d\boldsymbol{Y}_{i,p} + \sum_{j=1}^{d} \boldsymbol{Y}_{j,p} \\ &\geq -2d - 2d \\ &\geq -4d.\end{aligned}$$

Therefore, a matrix $\boldsymbol{A} = 4d\boldsymbol{J} + \boldsymbol{X}$ is guaranteed to have all entries at least 0, and further, the sum of each row and column must be exactly $4d^2$.

Now, let the entry $\boldsymbol{A}_{i,r}$ be the number of voters who place candidate i at rank r. Since $\boldsymbol{A}$ is a nonnegative integer matrix whose row and column sums are all $4d^2$, it can be written as the sum of $4d^2$ permutation matrices. Each permutation matrix describes one ranking over d candidates, as a value of 1 in the entry (i, p) indicates candidate i is placed at rank p. The permutation matrices can be found in time polynomial in d by repeatedly finding a perfect matching in a bipartite multigraph with $\boldsymbol{A}_{i,p}$ indicating the number of edges between candidate i and rank p, and thus the set of rankings S can be constructed in time polynomial in d.

Recall that we defined the matrix Δ where $\Delta_{i,p} := |i - p|$. Then the Spearman footrule cost matrix of placing candidate i at rank p given the input set S can be written as

$$\boldsymbol{C} = \boldsymbol{A}\Delta.$$

We now verify that $\boldsymbol{C}$ and $\boldsymbol{W}$ indeed have the same set of minimum cost perfect matchings when they are used as the cost of edges in a complete bipartite graph.

$$\begin{aligned}\boldsymbol{Q}\boldsymbol{C}\boldsymbol{Q} &= \boldsymbol{Q}\boldsymbol{A}\Delta\boldsymbol{Q} \\ &= \boldsymbol{Q}(4d\boldsymbol{J} + \boldsymbol{X})\Delta\boldsymbol{Q} \\ &= \boldsymbol{Q}\boldsymbol{X}\Delta\boldsymbol{Q} && \text{(As } \boldsymbol{Q}\mathbf{1} = 0\text{)} \\ &= -d\boldsymbol{Q}\boldsymbol{Q}\boldsymbol{W}\boldsymbol{L}_d\Delta\boldsymbol{Q} && \text{(By definition of } \boldsymbol{X}\text{)} \\ &= -d\boldsymbol{Q}\boldsymbol{W}\boldsymbol{L}_d\Delta\boldsymbol{Q} && \text{(As } \boldsymbol{Q}^2 = \boldsymbol{Q}\text{)} \\ &= 2d\boldsymbol{Q}\boldsymbol{W}\boldsymbol{Q}. && \text{(By Lemma G.8).}\end{aligned}$$

Therefore, we can write

$$\boldsymbol{Q}(\boldsymbol{C} - 2d\boldsymbol{W})\boldsymbol{Q} = 0.$$

Lemma G.7 now gives vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^d$ such that for every i, p,

$$\boldsymbol{C}_{i,p} = 2d\boldsymbol{W}_{i,p} + \boldsymbol{a}_i + \boldsymbol{b}_p.$$

For any perfect matching M from $[d]$ to $[d]$, this implies

$$\sum_{i=1}^{d} \boldsymbol{C}_{i,M(i)} = 2d \sum_{i=1}^{d} \boldsymbol{W}_{i,M(i)} + \sum_{i=1}^{d} \boldsymbol{a}_i + \sum_{p=1}^{d} \boldsymbol{b}_p.$$

The final two cost terms are identical and incurred by every perfect matching. Therefore, any matching M is a minimum cost perfect matching using $\boldsymbol{C}$ as the edge costs if and only if M is a minimum cost perfect matching using $\boldsymbol{W}$ as the edge costs. As observed at the beginning of the proof, M must then also be a perfect matching of the graph G. Hence, any optimal aggregate ranking $\sigma \in \mathcal{O}_S$ if and only if the corresponding perfect matching M is indeed a perfect matching of G. □

The construction in Lemma G.6 may produce a multiset S that contains duplicate rankings. Although a multiset is a valid input to the (diverse) rank aggregation problem, we show that S can be replaced by a set of distinct rankings on $2s$ additional candidates, preserving its optimal aggregate rankings (up to a fixed suffix) and the distances between them.

Lemma G.9. *Let S be a multiset of $s \geq 3$ rankings on d candidates. There is a set T of s distinct rankings on $d + 2s$ candidates and a ranking σ of the $2s$ new candidates such that for every optimal aggregate ranking π of S, the concatenation $\pi\sigma$ is an optimal aggregate ranking of T.*

Proof. Let $S = \{\sigma_1, \ldots, \sigma_s\}$ in any fixed order, and introduce $2s$ new candidates $t_1, \ldots, t_{2s}$. For each $\ell \in [s]$, let τ_ℓ be σ_ℓ followed by $t_1, \ldots, t_{2s}$ in order, except that $t_{2\ell-1}$ and $t_{2\ell}$ are swapped, and let $T = \{\tau_1, \ldots, \tau_s\}$. The rankings in T are distinct, since τ_ℓ is the only one that ranks $t_{2\ell}$ before $t_{2\ell-1}$. For a ranking π and a multiset R, write $\phi_R(\pi) = \sum_{\rho \in R} F(\pi, \rho)$.

Suppose a ranking places an original candidate c at position $q > d$ and a new candidate t_i at position $p \leq d$. In every τ_ℓ, c has some rank $a \leq d$ and t_i has some rank $b > d$. Since $p < q$ and $a < b$, swapping c and t_i reduces $F(\cdot, \tau_\ell)$ by

$$|q - a| + |p - b| - |p - a| - |q - b| = 2 \text{ length}([p, q] \cap [a, b]) \geq 2,$$

as the intersection contains $[d, d + 1]$. Summing over all s rankings, the swap reduces the total cost by at least $2s$, so the ranking is not optimal.

Therefore, new candidates occupy positions $d + 1, \ldots, d + 2s$ in any optimal ranking, so the total cost splits into a term depending only on the order of the original candidates and a term $\sum_i \sum_\ell |x_i - \tau_\ell(t_i)|$ depending only on the positions x_i of the new candidates. Each t_i has rank $d + i$ in $s - 1$ of the rankings and rank $d + i \pm 1$ in the remaining one. Since $s \geq 3$, $d + i$ is the unique median, so the inner sum is uniquely minimized at $x_i = d + i$, where it equals 1. These positions are distinct, so setting $x_i = d + i$ for all i is feasible and is the unique minimizer of the second term, with value $2s$ (see Dwork et al. (2001b)).

Let $\omega = t_1 \cdots t_{2s}$. The original candidates have the same ranks in τ_ℓ as in σ_ℓ, so for every ranking π of the original candidates,

$$\phi_T(\pi\omega) = \phi_S(\pi) + 2s.$$

where $\pi\omega$ is the ranking where π is followed by ω. Therefore, π is an optimal aggregate ranking of S if and only if $\pi\omega$ is an optimal aggregate ranking of T. Moreover, since all such rankings share the suffix ω, the footrule distance between any two of them equals the distance between their prefixes. □

G.4 Completing the Reduction

We now have all the tools to complete the reduction.

Proof of Theorem 4.1. Fix a constant $k \geq 3$. We use a reduction from k-diverse weighted perfect matching on bipartite graphs, which we have shown is NP-Hard from Lemma G.4 even for a k-regular bipartite graph. Take an instance of that problem with a k-regular weighted bipartite graph G with d vertices in each partition, and threshold T, where the maximum weight of any edge is $\text{poly}(d)$. Apply Lemma G.5 to obtain a new bipartite graph H over $\hat{d}$ candidates,

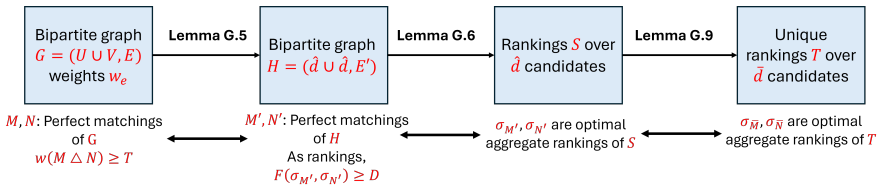


Figure 6: How our reductions map from k-diverse weighted perfect matchings to max-min diverse rank aggregation with fixed k.

the weight scale C, and the new threshold $D = CT$. Apply Lemma G.6 with H to obtain a list S of rankings over $\hat{d}$ candidates. Finally, we apply Lemma G.9 to obtain a set of unique rankings S' over d' candidates. The set S' and threshold D form the instance of max-min diverse rank aggregation for the fixed constant k.

Optimal aggregate rankings of S' are in bijection with perfect matchings of G by Lemmas G.5, G.6 and G.9. Under this bijection, the lemmas also give that for any two perfect matchings M_a, M_b on G, and the corresponding optimal aggregate rankings $\sigma_{M_a}, \sigma_{M_b} \in \mathcal{O}_{S'}$, it holds that

$$w(M_a \triangle M_b) \geq T \quad \iff \quad F(\sigma_{M_a}, \sigma_{M_b}) \geq D \tag{15}$$

Suppose we have a yes instance of k-diverse weighted perfect matchings on bipartite graphs. Let $\mathcal{T}$ be the satisfying set of perfect matchings; then for each matching $M_A \in \mathcal{T}$, construct its corresponding ranking over $\hat{d}$ candidates σ_{M_A}. The set $\mathcal{T}' = \{\sigma_{M_A} \mid M_A \in \mathcal{T}\}$ then forms the set of diverse max-min rankings. For any $M_A, M_B \in \mathcal{T}$, equation 15 gives us that $F(\sigma_{M_A}, \sigma_{M_B}) \geq D$, so $\mathcal{T}'$ is a satisfying solution to the max-min diverse rank aggregation instance. It is easy to see that the converse holds as well, so the reduction is valid.

See Figure 6 for a diagram on how the lemmas apply the reduction, and how the solutions map to each other.

As the weights of G are polynomially bounded by d for fixed k, the number of candidates $\hat{d}$ of H is polynomial with respect to d. Therefore, the constructed set of rankings S has size and number of candidates both polynomial in d. The number of output rankings stays equal to the fixed constant k throughout.

Membership in NP is straightforward. Let the certificate consist of k rankings. Compute the optimal rank aggregation cost by the standard reduction to minimum-cost bipartite perfect matching. Then verifying that every ranking in the certificate has this objective cost, and verifying all $\binom{k}{2}$ pairs of rankings in the certificate have distance at least D is clearly polynomial-time. □

We also get as a direct corollary that no fully polynomial-time approximation scheme (FPTAS) exists for the max-min diverse rank aggregation problem.

Corollary G.10. *Unless P=NP, the max-min diverse rank aggregation problem for any fixed $k \geq 3$ has no FPTAS.*

Proof. For rankings over d candidates, the Spearman footrule distance between any two rankings is an integer in the range $[0, \frac{d^2}{2}]$. Therefore, for any instance of the problem, $\texttt{OPT}_{min}$ must be an integer in this range.

Assume for the sake of contradiction that an FPTAS exists. That is, an algorithm with returns a set T such that $\texttt{Obj}_{min}(T) \geq (1-\epsilon)\texttt{OPT}_{min}$. Run this algorithm with $\epsilon = \frac{1}{2d^2}$.

Observe then that it must hold that $\texttt{OPT}_{min} - \texttt{Obj}_{min}(T) \geq 0$. Also as T is the output of the FPTAS with $\epsilon = \frac{1}{2d^2}$, it holds that

$$\begin{aligned}
\texttt{Obj}_{min}(T) &\geq (1 - \frac{1}{2d^2})\texttt{OPT}_{min} \\
\texttt{OPT}_{min} - \texttt{Obj}_{min}(T) &\leq \frac{1}{2d^2}\texttt{OPT}_{min} \\
&\leq \frac{1}{2d^2} \cdot \frac{d^2}{2} \\
&< 1.
\end{aligned}$$

As $\texttt{Obj}_{min}(T)$ and $\texttt{OPT}_{min}$ must be integers, it holds that $\texttt{Obj}_{min}(T) = \texttt{OPT}_{min}$. As $\epsilon = \frac{1}{2d^2}$, any FPTAS which runs in time polynomial in the input and in $\frac{1}{\epsilon}$ yields an exact polynomial time algorithm. This contradicts Theorem 4.1 unless P=NP. □

H Hardness of Diameter Among Approximate Medians

For a set of edges M, we again denote the sum of its weights for some weight function w using $w(M) = \sum_{e \in M} w_e$.

Here, we will prove that even max-diameter is NP-Complete when the diversity objective is over all rankings in $\mathcal{A}_\alpha$ for $\alpha > 0$. A significant amount of the reduction and the overall idea overlaps with Section G. First, recall that we have proven that the problem of extending a k-edge coloring on a k-regular bipartite graph is NP-Hard by Lemma G.2.

Lemma H.1. *Fix an integer $q \geq 3$. Given a q-regular bipartite graph H with nonnegative integer edge weights w and a positive integer B, it is NP-hard to decide whether the edges of H, $E(H)$, can be partitioned into perfect matchings $M_1, \ldots, M_q$ with $w(M_j) = B$ for every j. This remains true when*

$$w(E(H)) = qB \qquad \text{and} \qquad 0 \leq w_e \leq B \text{ for every edge } e,$$

and B is bounded by a polynomial in the size of H. Moreover, let β be the minimum weight of a perfect matching of H. Then $\beta \leq B$, and if $\beta = B$, the instance is a YES instance.

Proof. The proof is similar to that of Lemma G.4, and for brevity, we omit some of the identical details of that proof.

Fix some $q \geq 3$. Consider an instance of extending a q-edge coloring on a q-regular bipartite graph G. For each color j, let p_j be the number of edges of G precolored j, and define

$$p = \sum_{j=1}^{q} p_j, \qquad r = p + 1, \qquad d_j = p_j r^{j-1}, \qquad B = \sum_{j=1}^{q} d_j.$$

Form a graph H by adding a disjoint copy of $K_{q,q}$ to G. Choose the first vertex on the left of this new component and label its incident edges such that its edge to the first vertex on the right is f_1, the edge to the second vertex is f_2, and so on.

Let n be the number of vertices on each side of H. The target weight of matchings is the value B defined above.

Now, we give the edges the following weights in H.

$$w_e = \begin{cases} 0 & e \text{ is not precolored in } G, \\ r^{j-1} & e \text{ is precolored } j \text{ in } G, \\ B - d_j & e \text{ is the special edge } f_j. \end{cases}$$

The resulting graph H is q-regular and bipartite, and the total weight of all the edges is clearly qB, as for any color j, the weight of precolored edges and the special edge f_j sums to B. Also, weights clearly lie in $[0, B]$.

Now suppose that we have a yes instance of extending an edge precoloring of G. Take such a feasible coloring that extends it. Since G is q-regular, for any color j, all edges of color j form a perfect matching of G. Now, we color the additional $K_{q,q}$ such that f_j is colored j, and color the rest of the edges to achieve a valid edge coloring.

Consider the weight of the perfect matching from taking all edges of color j. The total weight must be equal to B. Since the matchings obtained must clearly be disjoint and cover all edges, we have a partitioning of the edges of H into q perfect matchings such that the weight is equal to B for every matching.

Conversely, take a partitioning of the edges of H into perfect matchings. A perfect matching of H must contain exactly one of the edges f_j, so label the matching that contains f_j as M_j. The remainder of the weight of M_j is $d_j = p_j r^{j-1}$ and must come from precolored edges.

Observe that by choice of the weights r^{j-1} for all edges of color j, the only possible way for M_j to have weight of d_j from precolored edges is to contain all edges precolored j. Now, color edges of G color j if they are in M_j. We must obtain a valid extension of the edge coloring of G.

The graph H is obtained from G by adding q^2 edges, and every matching used in the construction can be computed in polynomial time, so the reduction is polynomial. Moreover, since q is fixed,

$$B \leq pr^{q-1} = p(p+1)^{q-1},$$

which is bounded by a polynomial in the number of edges of G. Hence the problem is NP-hard.

Note that if $\beta = B$, where β is the minimum cost of any perfect matching of H, then the instance must be a yes instance, as a q-regular bipartite graph can be partitioned into q perfect matchings; and a perfect matching must have weight at least B, while the total weight of the edges is qB. □

In Lemma G.6, we constructed a set of input rankings S such that the optimal aggregate rankings were exactly in bijection with matchings using only edges of the tight edge graph. However, we now need to modify it slightly as we now work over approximate aggregate rankings; we will also adjust the order of exposition slightly due to the structure of the proof.

Lemma H.2. *For a multiset M of rankings, let $\Phi_M(\sigma) = \sum_{\rho \in M} F(\sigma, \rho)$ and $C^\star_M = \min_\sigma \Phi_M(\sigma)$. Let S be a multiset of $s \geq 3$ rankings of $[d]$. In time polynomial in d and s, one can construct a set R of s distinct rankings of $d + 2s$ candidates and a ranking ω of the $2s$ new candidates with the following properties.*

1. *For every ranking π of $[d]$, appending ω to π gives a ranking of Φ_R-cost equal to $\Phi_S(\pi) + 2s$.*
2. *$C^\star_R = C^\star_S + 2s$.*
3. *Every ranking of the $d + 2s$ candidates that does not end with ω has Φ_R-cost at least $C^\star_R + 2(s-2)$.*

Moreover, appending ω preserves footrule distances.

Proof. Let $S = \{\rho_1, \ldots, \rho_s\}$ and introduce $2s$ new candidates, called *tags*, numbered $1, \ldots, 2s$. Let ω be the ranking of the tags that places tag x at position $d + x$, and for a ranking π of $[d]$, write $\pi\omega$ for π followed by ω. For each $\ell \in [s]$, let τ_ℓ be ρ_ℓ followed by ω with tags $2\ell - 1$ and 2ℓ swapped, and let $R = \{\tau_1, \ldots, \tau_s\}$. The rankings in R are distinct, since only τ_ℓ ranks tag 2ℓ above tag $2\ell - 1$, and R is computable in polynomial time. Each tag occupies its position in ω in $s - 1$ rankings of R and an adjacent position in the remaining one.

Let π be a ranking of $[d]$. The original candidates occupy the same positions in τ_ℓ as in ρ_ℓ, so they contribute $\Phi_S(\pi)$ to $\Phi_R(\pi\omega)$, and each tag contributes 1. Hence $\Phi_R(\pi\omega) = \Phi_S(\pi) + 2s$.

We next show that every ranking σ of the $d + 2s$ candidates that does not end with ω satisfies

$$\Phi_R(\sigma) \geq C^\star_S + 2s + 2(s-2). \tag{16}$$

Together with $\Phi_R(\pi\omega) = \Phi_S(\pi) + 2s$, this gives $C^\star_R = C^\star_S + 2s$.

Suppose first that σ places the original candidates in the first d positions but the tags in an order other than ω. The original candidates contribute at least $C^\star_S$. A tag at its position in ω contributes 1, while a tag at any other position is misplaced in $s - 1$ rankings and contributes at least $s - 1$. At least $m \geq 2$ tags are misplaced, so the tags contribute at least $m(s-1) + (2s - m) = 2s + m(s-2) \geq 2s + 2(s-2)$.

Suppose instead that σ places some tag at a position $p \leq d$. Then some original candidate is at a position $q > d$. In every τ_ℓ, that candidate is at some $a \leq d$ and that tag at some $b > d$, so swapping the two in σ lowers $F(\sigma, \tau_\ell)$ by

$$|q - a| + |p - b| - |p - a| - |q - b| = 2\left|[p, q] \cap [a, b]\right| \geq 2,$$

since both intervals contain $[d, d+1]$. Each such swap therefore lowers Φ_R by at least $2s$. Repeating until the original candidates occupy the first d positions yields a ranking of cost at least $C^\star_S + 2s$, by $\Phi_R(\pi\omega) = \Phi_S(\pi) + 2s$ if it ends with ω and by the previous case otherwise. Hence $\Phi_R(\sigma) \geq C^\star_S + 4s > C^\star_S + 2s + 2(s-2)$.

Finally, $\pi\omega$ and $\pi'\omega$ place every tag at the same position, so $F(\pi\omega, \pi'\omega) = F(\pi, \pi')$. □

Recall that for a list S of rankings with optimal aggregation cost $C^\star_S$, $\mathcal{A}_\alpha := \{\sigma \in \mathcal{S}_d : \sum_{\pi \in S} F(\sigma, \pi) \leq (1+\alpha) C^\star_S\}$.

Lemma H.3. *Let $d \geq 2$, and let $\boldsymbol{W}$ be a $d \times d$ matrix of nonnegative integer assignment costs where $\min_{\sigma \in \mathcal{S}_d} \sum_{i \in [d]} \boldsymbol{W}_{i,\sigma(i)} = \beta$, and an integer $B > \beta$ be given.*

In time polynomial in d, $B+1$ and $\max_{i,p} \boldsymbol{W}_{i,p}$, one can construct a set R of distinct rankings and a rational $0 < \alpha < 1$ such that for all $\sigma \in \mathcal{A}_\alpha(R)$, it holds that $\sum_{i \in [d]} \boldsymbol{W}_{i,\sigma(i)} \leq B$.

Proof. Our construction will be similar to that of Lemma G.6, so for brevity, we omit some details of the calculations discussed in detail there.

Let

$$P = \max\left\{B+1, \max_{i,p} W_{i,p}\right\},$$

Recall the following notations from Lemma G.6. Let $\mathbf{1} \in \mathbb{R}^d$ be the all-ones vector, $\boldsymbol{J} = \mathbf{1}\mathbf{1}^\mathsf{T}$ be the all-ones matrix, and define the centering matrix $\boldsymbol{Q} = \boldsymbol{I} - \frac{1}{d}\boldsymbol{J}$. We also let Δ where $\Delta_{r,p} = |r-p|$ be the matrix of footrule distances, and let denote Laplacian matrix of the path on $[d]$ as $\boldsymbol{L}_d$.

$$\boldsymbol{X} = -d\boldsymbol{Q}\boldsymbol{W}\boldsymbol{L}_d, \qquad A = 8dP\boldsymbol{J} + \boldsymbol{X}.$$

Recall that if we define $\boldsymbol{Y} = \boldsymbol{W}\boldsymbol{L}_d$, the entries of $\boldsymbol{Y}$ take the second differences along rows of $\boldsymbol{W}$. So for any i, p we have $|Y_{i,p}| \le 2P$, and

$$\boldsymbol{X}_{i,p} = -d\boldsymbol{Y}_{i,p} + \sum_{j=1}^{d} \boldsymbol{Y}_{j,p}.$$

Hence entries of $\boldsymbol{X}$ are integral and $\boldsymbol{X}_{i,p} \ge -4dP$. Therefore, we can conclude that all entries of $\boldsymbol{A}$ are non-negative.

Moreover, every row and column of $\boldsymbol{X}$ sum to 0, so every row and column of $\boldsymbol{A}$ sums to $8d^2P$.

As every row and every column of $\boldsymbol{A}$ sum to $8d^2P$, it can be written as the sum of $8d^2P$ permutation matrices, which can be obtained via repeatedly finding perfect matchings in the bipartite multigraph induced by $\boldsymbol{A}$, each of which represents one ranking. The set of all such rankings obtained forms the set of rankings S.

In the matrix $\boldsymbol{A}\Delta$, the entry (i,p) gives the contribution to the Spearman footrule cost of placing candidate i at rank p. Using the fact that $\boldsymbol{L}\Delta\boldsymbol{Q} = -2\boldsymbol{Q}$ from Lemma G.8,

we obtain

$$\boldsymbol{Q}(\boldsymbol{A}\Delta)\boldsymbol{Q} = \boldsymbol{Q}\boldsymbol{X}\Delta\boldsymbol{Q} = 2d\boldsymbol{Q}\boldsymbol{W}\boldsymbol{Q}.$$

By Lemma G.7, it follows that there exist vectors $\boldsymbol{a}, \boldsymbol{b}$ such that

$$(\boldsymbol{A}\Delta)_{i,p} = 2dW_{i,p} + a_i + b_p.$$

Then for any ranking σ, the rank aggregation objective cost of σ with respect to S is

$$\sum_{i\in[d]} \boldsymbol{W}_{i,\sigma(i)} + \sum_{i\in[d]} \boldsymbol{a}_i + \sum_{i\in[d]} \boldsymbol{b}_i.$$

We have $s = 8d^2P$ rankings in S. Now, we apply the construction of Lemma H.2 with the rankings S to obtain a set of distinct rankings R and ranking ω over the $2s$ new candidates. Observe that by Lemma H.2, we have that $C^\star_R = C^\star_R + 2s$. Now, we show a value of α to prove the claim, for rankings in $\mathcal{A}_\alpha(R)$.

We have that

$$C^\star_S = 2d\beta + \sum_{i\in[d]} \boldsymbol{a}_i + \sum_{i\in[d]} \boldsymbol{b}_i.$$

Therefore, we can lower-bound $C^\star_S$ and $C^\star_R$. Since every entry of A is at least $4dP$, every candidate of $[d]$ contributes at least $4dP(d-1)$ to the cost of any aggregate ranking. Therefore

$$C^\star_R > C^\star_S \ge 4d^2P(d-1) > 2dB.$$

Define

$$\alpha = \frac{2d(B-\beta)}{C^\star_R}.$$

Because $\beta < B$, we have $\alpha > 0$, while the preceding lower bound gives

$$\alpha \le \frac{2dB}{C^\star_R} < 1.$$

We now argue that any ranking in $\mathcal{A}_\alpha(R)$ must be of the form $\sigma\omega$ for some ranking $\sigma \in \mathcal{S}_d$. By Lemma H.2 point 3, any ranking that does not have the form $\sigma\omega$ has cost at least $C^\star_S + 4s - 2$. Further, we have that $s = 8d^2P$, and that $P > B$. Therefore, suppose that a ranking $\sigma' \in \mathcal{A}_\alpha(R)$ is not of form $\sigma\omega$. Then we have that its aggregation cost is

$$\begin{aligned}\phi_R(\sigma') &\geq C_S^\star + 4s - 2 \\ &\geq C_S^\star + 2s + 16d^2B - 2 \\ &> C_S^\star + 2s + 2dB \\ &\geq (1+\alpha)C_R^\star && \text{(By the choice of } \alpha.\text{)}\end{aligned}$$

Therefore, any ranking $\sigma \in \mathcal{A}_\alpha(R)$ must place the original d candidates in the first d ranks, and therefore we have the following.

$$\begin{aligned}\sigma \in \mathcal{A}_\alpha(R) &\iff \sum_{\pi \in R} F(\pi, \sigma) \leq (1+\alpha)C_R^\star \\ &\iff \sum_{i\in[d]} \boldsymbol{a}_i + \sum_{i\in[d]} \boldsymbol{b}_i + 2d \sum_{i\in[d]} \boldsymbol{W}_{i,\sigma(i)} + 2s \leq \sum_{i\in[d]} \boldsymbol{a}_i + \sum_{i\in[d]} \boldsymbol{b}_i + 2d\beta + 2s + 2d(B-\beta) \\ &\iff 2d \sum_{i\in[d]} \boldsymbol{W}_{i,\sigma(i)} \leq 2dB \\ &\iff \sum_{i\in[d]} \boldsymbol{W}_{i,\sigma(i)} \leq B.\end{aligned}$$

The set R and value α as described prove the claim.

The number of input rankings is polynomial in d and P, while the number of candidates remains as d. Therefore, this construction is polynomial time in the size of the input. □

Now, we have the tools to prove Theorem 5.1, that approximate-median max-diameter rank aggregation is NP-Complete.

Proof of Theorem 5.1. We use a reduction from an instance of partitioning a weighted 3-regular bipartite graph G into three perfect matchings of target weight B, which we have shown is NP-Hard by Lemma H.1. We will assume the minimum cost perfect matching on G has cost $\beta < B$; otherwise, trivially we have a yes instance. Take such an instance graph G with d vertices on each partition, where edges have weight $\mathrm{poly}(d)$, and an input target B.

We will now characterize the three matchings using only two of them. Consider perfect matchings with weight at most B. Define a new weight function $w'_e = 1 + w_e$, and $T = 2d + 2B$. For any two perfect matchings M, N satisfying the instance,

$$\begin{aligned}w'(M \triangle N) &= 2d - 2|M \cap N| + w(M) + w(N) - 2w(M \cap N) \\ &\leq 2d + 2B\end{aligned}$$

Observe that equality holds only when $|M \cap N| = 0$ and $w(M) = w(N) = B$. Then consider all of the edges not in M or N. They must form a perfect matching of weight (under w) of $3B - B - B = B$. Thus, if a pair of perfect matchings M, N have $w(M) \leq B, w(N) \leq B$ and have $w'(M \triangle N) \geq T$, then the edges of G can be partitioned into three perfect matchings of weight (under w) exactly B. The converse holds, as if the edges of G can be partitioned into three perfect matchings of weight (under w) B, then taking any two gives a weight on their symmetric difference under w' of $2d + 2B$. Therefore, the edges of G can be partitioned into three perfect matchings of weight exactly B if and only if there are perfect matchings M, N with $w(M) \leq B, w(N) \leq B$, and $w'(M \triangle N) \geq 2d + 2B$.

Now, we will again use Lemma G.5 to construct a bipartite graph where the vertex sets represent candidates and ranks. Use Lemma G.5 with G as input with edge weights w', producing a bipartite graph H, dimension $\hat{d}$, and C. Now, construct a matrix $\boldsymbol{W}$ of size $\hat{d} \times \hat{d}$, where entry $\boldsymbol{W}_{i,p}$ will represent the cost of assigning candidate i to rank p.

Consider a path constructed from an edge $e = (u_i, v_j)$ with weight w_e of G. The construction of H creates a path of length 5 starting with the edge $(2i, p_e)$. Set the entry of $\boldsymbol{W}_{2i,p_e} = w_e$. For all other edges of H of form $e = (u, v)$, set entry $\boldsymbol{W}_{u,v} = 0$. For all pairs of candidates, ranks i, p which have no edge in H, set $\boldsymbol{W}_{i,p} = B + 1$.

Now, using $\boldsymbol{W}_{i,p}$ as the cost of placing candidate i at rank p, any assignment of candidates to ranks with cost at most B must use only edges in H. We can then form a bijection to a perfect matching M of G; if the assignment places candidate $2i$ at rank p_e, and the constructed path ends with rank $2j$, then M has the edge $e = (u_i, v_j)$. Conversely,

every matching of weight at most B in G can be mapped to such an assignment of candidates to ranks by following the construction. Any assignment assigning a candidate to a rank where no edge exists in H costs at least $B + 1$.

Apply Lemma H.3 with $\boldsymbol{W}$ and value B, and obtain a list of rankings S and rational α. The set S, value α and the diversity threshold $D = CT = C(2d + 2B)$ form the instance of $(1 + \alpha)$-approximate diameter diverse rank aggregation.

Each ranking in $\mathcal{A}_\alpha$ is bijective to matchings of G with weight at most B. Suppose we have a yes instance of $(1 + \alpha)$-approximate diameter diverse rank aggregation. Let the two rankings be $\sigma_M, \sigma_N \in \mathcal{A}_\alpha$, and corresponding by the bijection to perfect matchings M, N of G. We have that $w(M) \leq B$ and $w(N) \leq B$. Further, as σ_M and σ_N use place candidates at ranks only if such an edge exists in H, we have that

$$F(\sigma_M, \sigma_N) \geq D \quad \iff \quad w'(M \triangle N) \geq T.$$

Therefore, this implies that $w(M) = w(N) = B$, and taking all edges not in M or N gives a perfect matching of weight exactly B, as discussed earlier. Therefore, a yes instance of $(1 + \alpha)$-approximate diameter diverse rank aggregation implies a yes instance of equal weight partitioning the 3-regular bipartite graph G. The converse is straightforward, by taking any two matchings M, N where $w(M) = w(N) = B$ and following the constructions to obtain two rankings $\sigma_M, \sigma_N \in \mathcal{A}_\alpha$.

All graph weights and all sizes of the instances in the construction are polynomially bounded in the length of the input graph G.

For membership in NP, compute $C^\star_S$ by minimum cost bipartite matching, then verify the two certificate rankings have rank aggregation objective cost at most $(1 + \alpha)C^\star$ and that their footrule distance is at least D. □

I The Generalized Assignment Oracle

For each candidate i and position p, as before, let $c_{i,p} = \sum_{\pi \in S} |p - \pi(i)|$ be the contribution to the rank aggregation objective cost of assigning candidate i to rank p.

Proof of Lemma 5.2. Introduce a variable $r_{i,p}$ for every edge, a variable y and a variable z_j for each $j \in [t]$ and form a matrix $\boldsymbol{A}$ of size $d \times d$ where

$$\boldsymbol{A}_{i,p} = r_{i,p} y^{c_{i,p}} \prod_{j-1}^{t} z_j^{w^j_{i,p}}.$$

The variable y will track the rank aggregation objective cost, while the variables z_j will track the sum of labels for a ranking σ. The edge variables avoid cancellations in the determinant. If any ranking $\sigma \in \mathcal{S}_d$ has a rank aggregation objective cost of C, and label vector $(w_1, \ldots, w_t)$, in the determinant it contributes the following term

$$\prod_{i=1}^{d} \left(r_{i,\sigma(i)} y^{c_{i,\sigma(i)}} \prod_{j=1}^{t} z_j^{w^j_{i,\sigma(i)}} \right) = y^C \prod_{j=1}^{t} z_j^{w_j} \prod_{i=1}^{d} r_{i,\sigma(i)}.$$

Distinct rankings contribute distinct products on the edge variables r, thus avoiding cancellations in the symbolic determinant after summation. Therefore, if there is a ranking with rank aggregation objective C and label vector $(w_1, \ldots, w_t)$, the monomial $y^C \prod_{j=1}^{t} z_j^{w_j}$ has a non-zero coefficient and appears in the determinant. The degree of y is at most nB_d, while there are at most $(W_j + 1)^t$ many distance vectors $\boldsymbol{w}$ when the coordinate of any entry of the distance vector is upper bounded by W_j.

Take a finite field $\mathbb{F}$ where

$$|\mathbb{F}| > \frac{nB_d(W_j + 1)^t}{\delta}$$

Now, sample elements uniformly at random from $\mathbb{F}$ to substitute for each edge variable $r_{i,p}$, and then compute the determinant of the matrix after this substitution. The Schwartz–Zippel lemma and a union bound over at most $nB_d(W_j + 1)^t$ possible combinations of the rank aggregation objective and label vectors ensure that all label vectors attained by rankings which are in $\mathcal{A}_\alpha$ are nonzero with probability at least $1 - \delta$.

Now, we can inspect all monomials where the degree of y is at most $(1+\alpha)C^\star$, and find the monomial which maximizes the function Φ over the degrees of z. Then the ranking that attains this can be found via self-reduction as in Lemma E.5.

If the determinant is 0, compute and return an exact median using reduction to minimum cost perfect matching to output a feasible solution.

There are now $(nB_d+1)(W_j+1)^t$ possible distinct monomials in the determinant; therefore, using a standard determinant computation algorithm requires time polynomial in $(nB_d+1)(W_j+1)^t$ and d to compute the determinant. The self-reduction adds a $\mathrm{poly}(d)$ factor in terms of the number of determinant computations. □

Let us consider some examples to understand the use of this oracle, with given rankings $\sigma_1,\ldots,\sigma_t$. Setting the values of w as

$$w^j_{i,p} = |p-\sigma_j(i)|, \qquad j\in[t],$$

and setting Φ as the $\min$ function gives a randomized exact robust-farthest oracle over $\mathcal{A}_\alpha$ in polynomial time when t is fixed. The values of $w^j_{i,p}$ now track the contribution to the Spearman footrule distance from placing candidate i at rank p to σ_j, and Φ outputs the minimum distance to any ranking of $\sigma_1,\ldots,\sigma_j$.

Alternatively, consider we also are given integer weights $\lambda_1,\ldots,\lambda_t$, using a single dimensional w as

$$w_{i,p} = \sum_{j\in[t]} \lambda_j |p-\sigma_j(i)|$$

and setting Φ as the identity function maximizes the weighted sum of distances $\sum_{j\in[t]}\lambda_j F(\sigma,\sigma_j)$, as the single value w now tracks the weighted sum of distances to $\sigma_1,\ldots,\sigma_j$.

J Approximation for Approximate-Median Diameter

Theorem J.1. *For every rational $\alpha\geq 0$ and failure probability $0<\zeta<1$, there is a randomized polynomial-time algorithm that returns $\sigma,\tau\in\mathcal{A}_\alpha$ satisfying $F(\sigma,\tau)\geq\frac{1}{2}OPT_{\mathrm{diam}}(\mathcal{A}_\alpha)$ with probability at least $1-\zeta$. Both σ and τ are guaranteed to be in $\mathcal{A}_\alpha$ even if the randomized oracle fails.*

Proof. Compute an exact median σ_0, which belongs to $\mathcal{A}_\alpha$, and use Lemma 5.2 with the single label $|p-\sigma_0(i)|$, with failure probability set to ζ. With just one label, with Φ as the identity function, the oracle of Lemma 5.2 finds a ranking in $\mathcal{A}_\alpha$ that is farthest from σ_0. Suppose that $\sigma^\star,\sigma'\in\mathcal{A}_\alpha$ are two rankings that give an optimal diameter.

Then it holds that

$$\begin{aligned} F(\sigma^\star,\sigma') &\leq F(\sigma^\star,\sigma_0)+F(\sigma_0,\sigma') && \text{(By triangle inequality)}\\ &\leq 2F(\tau,\sigma_0) && \text{(As } \tau \text{ is the farthest ranking in } \mathcal{A}_\alpha \text{ from } \sigma_0)\end{aligned}$$

This proves the claim. □

K Negative-Type Analysis for Approximate-Median Max-Sum Diversity

The approximate-median max-sum diversity analysis uses the negative-type geometry of the Spearman footrule; therefore, we provide a proof of it for completeness.

Lemma K.1. *Let $T=(\sigma_1,\ldots,\sigma_t)$ be any finite collection of rankings, and let $z_1,\ldots,z_t$ be real values satisfying $\sum_{i\in[t]} z_i = 0$. Then*

$$\sum_{i,j\in[t]} z_i z_j F(\sigma_i,\sigma_j)\leq 0.$$

Consequently, let p and q be probability distributions on $\mathcal{A}_\alpha$, and define

$$f(p) := \frac{1}{2}\mathbb{E}_{X,X'\sim p}\big[F(X,X')\big],$$

where X and X' are drawn independently from p. Then

$$\mathbb{E}_{X\sim p,\,Y\sim q}\big[F(X,Y)\big]\geq f(p)+f(q). \tag{17}$$

Here, all random draws in the displayed expectations are independent.

Proof. We first express the footrule distance as a sum of binary threshold distances. For every candidate $a \in [d]$ and threshold $h \in [d-1]$, define

$$\phi_{a,h}(\sigma) := \mathbf{1}\{\sigma(a) > h\}.$$

Thus, $\phi_{a,h}(\sigma)$ records whether ranking σ places candidate a after position h.

For any two positions $a, b \in [d]$, we have $|a-b| = \sum_{h=1}^{d-1} |\mathbf{1}\{a > h\} - \mathbf{1}\{b > h\}|$.

Applying this identity to any two rankings σ_i, σ_j gives

$$\begin{aligned} F(\sigma_i, \sigma_j) &= \sum_{a=1}^{d} |\sigma_i(a) - \sigma_j(a)| \\ &= \sum_{a=1}^{d} \sum_{h=1}^{d-1} |\phi_{a,h}(\sigma_i) - \phi_{a,h}(\sigma_j)|. \end{aligned} \tag{18}$$

For any fixed a, h, as the values of ϕ are in $\{0,1\}$, we have for any σ_i, σ_j

$$|\phi_{a,h}(\sigma_i) - \phi_{a,h}(\sigma_j)| = \phi_{a,h}(\sigma_i) + \phi_{a,h}(\sigma_j) - 2\phi_{a,h}(\sigma_i)\phi_{a,h}(\sigma_j).$$

Therefore,

$$\begin{aligned} &\sum_{i,j\in[t]} z_i z_j |\phi_{a,h}(\sigma_i) - \phi_{a,h}(\sigma_j)| \\ &= \sum_{\sigma_i,\sigma_j} z_i z_j \phi_{a,h}(\sigma_i) + \sum_{\sigma_i,\sigma_j} z_i z_j \phi_{a,h}(\sigma_j) - 2\sum_{\sigma_i,\sigma_j} z_i z_j \phi_{a,h}(\sigma_i)\phi_{a,h}(\sigma_j) \\ &= \left(\sum_{j\in[t]} z_j\right)\left(\sum_{i\in[t]} z_i \phi_{a,h}(\sigma_i)\right) + \left(\sum_{i\in[t]} z_i\right)\left(\sum_{j\in[t]} z_j \phi_{a,h}(\sigma_j)\right) - 2\left(\sum_{i\in[t]} z_i \phi_{a,h}(\sigma_i)\right)^2. \end{aligned}$$

The first two terms vanish because $\sum_{i\in[t]} z_i = 0$. Hence

$$\sum_{i,j\in[t]} z_i z_j |\phi_{a,h}(\sigma_i) - \phi_{a,h}(\sigma_j)| = -2\left(\sum_{i\in[t]} z_i \phi_{a,h}(\sigma_i)\right)^2 \le 0.$$

Using equation 18, we get

$$\begin{aligned} \sum_{i,j\in[t]} z_i z_j F(\sigma_i, \sigma_j) &= \sum_{a=1}^{d}\sum_{h=1}^{d-1}\sum_{i,j\in[t]} z_i z_j |\phi_{i,h}(\sigma_i) - \phi_{i,h}(\sigma_j)| \\ &= -2\sum_{a=1}^{d}\sum_{h=1}^{d-1}\left(\sum_{i\in[d]} z_i \phi_{i,h}(\sigma_i)\right)^2 \\ &\le 0. \end{aligned}$$

This proves that the Spearman footrule distance is of negative type.

It remains to derive equation 17. For every $\sigma \in \mathcal{A}_\alpha$, set

$$z_\sigma := p(\sigma) - q(\sigma).$$

Since p and q are probability distributions,

$$\sum_{\sigma\in\mathcal{A}_\alpha} z_\sigma = \sum_{\sigma} p(\sigma) - \sum_{\sigma} q(\sigma) = 1 - 1 = 0.$$

Applying the first part of the lemma to these coefficients gives

$$0 \ge \sum_{\sigma,\tau} \big(p(\sigma) - q(\sigma)\big)\big(p(\tau) - q(\tau)\big) F(\sigma, \tau).$$

Expanding the product and using the symmetry of F, we obtain

$$\begin{aligned} 0 &\geq \mathbb{E}_{X,X'\sim p}[F(X,X')] + \mathbb{E}_{Y,Y'\sim q}[F(Y,Y')] \\ &\quad - 2\mathbb{E}_{X\sim p,\, Y\sim q}[F(X,Y)] \\ &= 2f(p) + 2f(q) - 2\mathbb{E}_{X\sim p,\, Y\sim q}[F(X,Y)]. \end{aligned}$$

Rearranging proves

$$\mathbb{E}_{X\sim p,\, Y\sim q}[F(X,Y)] \geq f(p) + f(q),$$

as claimed. □

Proof of Theorem 5.4. Let

$$\Delta(\mathcal{A}_\alpha) := \left\{ \mu : \mathcal{A}_\alpha \to [0,1] \ : \ \sum_{\sigma\in\mathcal{A}_\alpha} \mu(\sigma) = 1 \right\}$$

denote the set of probability distributions over $\mathcal{A}_\alpha$. For a ranking $\sigma \in \mathcal{A}_\alpha$, let δ_σ denote the point-mass distribution at σ defined as,

$$\delta_\sigma(\tau) = \begin{cases} 1, & \tau = \sigma, \\ 0, & \tau \neq \sigma. \end{cases}$$

For $\mu \in \Delta(\mathcal{A}_\alpha)$, recall that

$$f(\mu) := \frac{1}{2}\mathbb{E}_{\sigma,\tau\sim\mu}[F(\sigma,\tau)],$$

where the two rankings are drawn independently from μ. Define

$$f^\star := \max_{\mu\in\Delta(\mathcal{A}_\alpha)} f(\mu). \tag{31}$$

Remark. The distribution attaining $f^\star$ is used only in the analysis and the algorithm does not need to compute it.

The distributional optimum upper-bounds the optimum over k-rankings. Let $T^\star = (\sigma_1^\star, \ldots, \sigma_k^\star)$ be an optimal k-tuple of rankings in $\mathcal{A}_\alpha$, and consider its empirical distribution

$$\mu_T^\star := \frac{1}{k}\sum_{a\in[k]} \delta_{\sigma_a^\star}.$$

Then

$$\begin{aligned} f(\mu_T^\star) &= \frac{1}{2k^2}\sum_{a,b\in[k]} F(\sigma_a^\star, \sigma_b^\star) \\ &= \frac{1}{k^2}\sum_{1\leq a<b\leq k} F(\sigma_a^\star, \sigma_b^\star) \\ &= \frac{1}{k^2}\mathtt{Obj}_{sum}(T^\star) = \frac{1}{k^2}\mathtt{OPT}_{\mathrm{sum}}^{\alpha}. \end{aligned}$$

Because $f^\star$ is the maximum over all distributions on $\mathcal{A}_\alpha$, this gives

$$\mathtt{OPT}_{\mathrm{sum}}^{\alpha} \leq k^2 f^\star. \tag{19}$$

Moreover, since $F(\sigma,\tau) \leq B_d$ for every pair of rankings,

$$f^\star \leq \frac{B_d}{2}. \tag{20}$$

Construction of the distributions. If $B_d = 0$, all pairwise footrule distances are zero and the result is immediate. Hence assume $B_d > 0$, and set $R := 4k^3 B_d$. Compute an optimal aggregate ranking $\sigma_1 \in \mathcal{A}_\alpha$, and initialize $\mu_1 := \delta_{\sigma_1}$. For each $t = 1, \ldots, R-1$, we use Lemma 5.2 to find a ranking

$$\sigma_{t+1} \in \arg\max_{\sigma\in\mathcal{A}_\alpha} \mathbb{E}_{\tau\sim\mu_t}[F(\sigma,\tau)], \tag{21}$$

and then update the distribution according to

$$\mu_{t+1} := \frac{t}{t+2}\mu_t + \frac{2}{t+2}\delta_{\sigma_{t+1}}. \tag{22}$$

An induction on t shows that

$$\mu_t = \frac{2}{t(t+1)} \sum_{j=1}^{t} j\, \delta_{\sigma_j}. \tag{23}$$

Indeed, the identity holds for $t = 1$. Assuming it holds for t, equation 22 gives

$$\begin{aligned}
\mu_{t+1} &= \frac{t}{t+2}\left(\frac{2}{t(t+1)} \sum_{j=1}^{t} j\, \delta_{\sigma_j}\right) + \frac{2}{t+2}\delta_{\sigma_{t+1}} \\
&= \frac{2}{(t+1)(t+2)}\left(\sum_{j=1}^{t} j\, \delta_{\sigma_j} + (t+1)\delta_{\sigma_{t+1}}\right) \\
&= \frac{2}{(t+1)(t+2)} \sum_{j=1}^{t+1} j\, \delta_{\sigma_j}.
\end{aligned}$$

Implementing each maximization with the assignment oracle. We now specify precisely how Lemma 5.2 implements equation 21. At iteration t, assign to every candidate–position pair (i, p) the single nonnegative integer label

$$w_{i,p}^{(t)} := \sum_{j=1}^{t} j\, |p - \sigma_j(i)|. \tag{24}$$

For any ranking σ, its total label value is

$$\begin{aligned}
\sum_{i\in[d]} w_{i,\sigma(i)}^{(t)} &= \sum_{i\in[d]} \sum_{j=1}^{t} j\, |\sigma(i) - \sigma_j(i)| \\
&= \sum_{j=1}^{t} j\, F(\sigma, \sigma_j) \\
&= \frac{t(t+1)}{2} \mathbb{E}_{\tau\sim\mu_t}[F(\sigma, \tau)],
\end{aligned} \tag{25}$$

where the final equality follows from equation 23. The factor $t(t+1)/2$ does not depend on σ, so maximizing the label total is equivalent to maximizing the expectation in equation 21.

Thus, at iteration t, we invoke Lemma 5.2 with $w(\sigma) = \sum_{i\in[d]} w_{i,\sigma(i)}^{(t)}$, $\Phi(v) = v$, and failure probability ζ/R, the oracle returns a ranking in $\mathcal{A}_\alpha$ such that with probability at least $1 - \zeta/R$ it maximizes $w(\sigma)$ and therefore by equation 23, maximizes the expectation in equation 21 as well.

Furthermore, since $F(\sigma, \sigma_j) \le B_d$ for every j, we have $w(\sigma) \le W_t := B_d \sum_{j=1}^{t} j = B_d \frac{t(t+1)}{2}$.

Progress in the distributional objective. Conditioned on all oracle calls being exact, let $f_t := f(\mu_t)$, and let $\mu^\star \in \arg\max_{\mu\in\Delta(\mathcal{A}_\alpha)} f(\mu)$, so that $f(\mu^\star) = f^\star$.

Applying Lemma K.1 to $\mu^\star$ and μ_t gives

$$\mathbb{E}_{\sigma\sim\mu^\star,\, \tau\sim\mu_t}[F(\sigma, \tau)] \ge f^\star + f_t. \tag{26}$$

Note that the left-hand side is an average over $\sigma \sim \mu^\star$, of $\mathbb{E}_{\tau\sim\mu_t}[F(\sigma, \tau)]$.

Therefore, at least one ranking in $\mathcal{A}_\alpha$ attains a value of at least $f^\star + f_t$. Since σ_{t+1} is an exact maximizer,

$$\mathbb{E}_{\tau\sim\mu_t}[F(\sigma_{t+1}, \tau)] \ge f^\star + f_t. \tag{27}$$

Expanding $f(\mu_{t+1})$ using equation 22, and using $F(\sigma_{t+1}, \sigma_{t+1}) = 0$, gives

$$\begin{aligned} f_{t+1} &= \left(\frac{t}{t+2}\right)^2 f_t + \frac{2t}{(t+2)^2}\mathbb{E}_{\tau\sim\mu_t}[F(\sigma_{t+1}, \tau)] \\ &\geq \left(\frac{t}{t+2}\right)^2 f_t + \frac{2t}{(t+2)^2}(f_t + f^\star) \\ &= \frac{t}{t+2} f_t + \frac{2t}{(t+2)^2} f^\star. \end{aligned}$$

Let $e_t := f^\star - f_t$. The preceding inequality implies

$$e_{t+1} \leq \frac{t}{t+2} e_t + \frac{4f^\star}{(t+2)^2}. \tag{28}$$

We claim that for every $t \geq 1$

$$e_t \leq \frac{4f^\star}{t+2}. \tag{29}$$

For $t = 1$, $\mu_1 = \delta_{\sigma_1}$, so $f_1 = 0$ and

$$e_1 = f^\star \leq \frac{4f^\star}{3}.$$

For the induction step, equation 28 and equation 29 give

$$\begin{aligned} e_{t+1} &\leq \frac{t}{t+2}\frac{4f^\star}{t+2} + \frac{4f^\star}{(t+2)^2} \\ &= \frac{4(t+1)f^\star}{(t+2)^2} \\ &\leq \frac{4f^\star}{(t+1)+2}, \end{aligned}$$

where the final inequality follows from $(t+1)((t+1)+2) \leq (t+2)^2$. Therefore, we can see that

$$f(\mu_R) \geq f^\star - \frac{4f^\star}{R+2}. \tag{30}$$

Rounding the final distribution. Draw $\tau_1, \ldots, \tau_k$ independently from μ_R. Their expected sum diversity is

$$\begin{aligned} \mathbb{E}\left[\sum_{1\leq a<b\leq k} F(\tau_a, \tau_b)\right] &= \binom{k}{2}\mathbb{E}_{\sigma,\tau\sim\mu_R}[F(\sigma, \tau)] \\ &= k(k-1)f(\mu_R). \end{aligned}$$

Using equation 19, equation 20, and equation 30, we obtain

$$\begin{aligned} k(k-1)f(\mu_R) &\geq k(k-1)f^\star - \frac{4k(k-1)f^\star}{R+2} \\ &\geq \frac{k-1}{k}\texttt{OPT}^\alpha_{\text{sum}} - \frac{2k(k-1)B_d}{R+2}. \end{aligned} \tag{31}$$

The support of μ_R is explicitly contained in $\{\sigma_1, \ldots, \sigma_R\}$. We derandomize the k independent draws by the method of conditional expectations. Namely, we fix $\tau_1, \ldots, \tau_k$ one at a time. When fixing τ_a, we try every ranking in the support of μ_R and select one for which the conditional expected final sum diversity does not decrease. After all k rankings have been fixed, we obtain a tuple

$$T = (\tau_1, \ldots, \tau_k)$$

satisfying

$$\texttt{Obj}_{sum}(T) \geq \frac{k-1}{k}\texttt{OPT}^\alpha_{\text{sum}} - \frac{2k(k-1)B_d}{R+2}. \tag{32}$$

Since $R = 4k^3 B_d$,

$$\frac{2k(k-1)B_d}{R+2} < \frac{1}{k}.$$

Since $\texttt{OPT}^{\alpha}_{\text{sum}} \in \mathbb{Z}$, we can see that $\frac{k-1}{k}\texttt{OPT}^{\alpha}_{\text{sum}} \in (1/k)\mathbb{Z}$, while $\texttt{Obj}_{sum}(T)$ is an integer. Therefore, an additive loss strictly smaller than $1/k$ cannot place $\texttt{Obj}_{sum}(T)$ below $\frac{k-1}{k}\texttt{OPT}^{\alpha}_{\text{sum}}$. Hence

$$\texttt{Obj}_{sum}(T) \geq \left(1 - \frac{1}{k}\right)\texttt{OPT}^{\alpha}_{\text{sum}}.$$

Success probability and running time. There are $R - 1 < R$ adaptive oracle calls. Conditioned on the complete history before any call, its failure probability is at most ζ/R. A union bound therefore implies that all calls are exact with probability at least $1 - \zeta$.

Note that Lemma 5.2 always returns a ranking in $\mathcal{A}_\alpha$ (even if an oracle call fails to be optimal). Thus every ranking in the support of μ_R, and consequently every ranking in the final tuple T, satisfies the aggregation-cost budget for every outcome of the randomized algorithm.

Finally, $R = 4k^3 B_d$ is polynomial in d and k, and every oracle call has one label coordinate bounded by

$$W_t = B_d \frac{t(t+1)}{2} \leq B_d \frac{R(R+1)}{2}.$$

Therefore, all oracle calls and the conditional-expectation derandomization run in polynomial time. □

L Approximation Algorithm for Max-Min Diversity for Approximate Medians

Proof of Theorem 5.5. We use a similar algorithm and proof as in Theorem 4.2 and Lemma E.5. First, find any optimal aggregate ranking σ_1, and set $T = \{\sigma_1\}$.

Now, we repeat the following for $t = 2, \ldots, k$. Apply the oracle of Lemma 5.2 with the value $w^j_{i,p} = |p - \sigma_j(i)|$ for all $j < t$, and set the function Φ as the $\min$ function; so the oracle maximizes the minimum distance. This finds a ranking σ_t of $\mathcal{A}_\alpha$ that maximizes the minimum distance to any ranking in T_{t-1}. Let $T_t = T_{t-1} \cup \{\sigma_t\}$. Finally, return T_k as the output set.

The oracle is called k times, and the dimension of the w vector is at most k, and each coordinate has value at most B_d, giving the claimed running time.

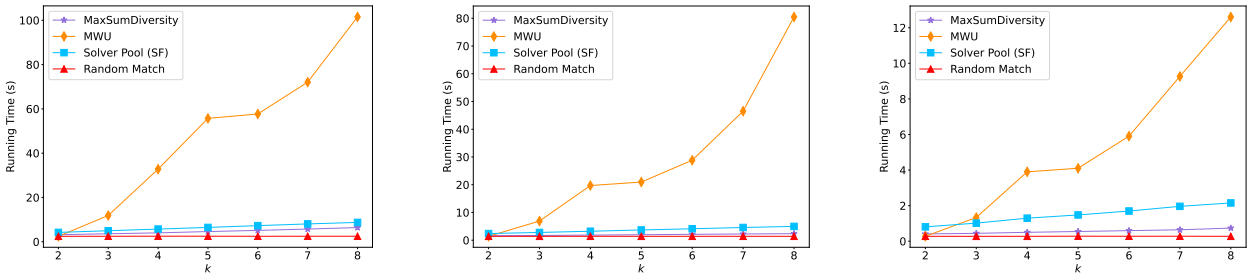
Consider balls of radius $\texttt{OPT}^{\alpha}_{\text{min}}/2$ centered at the rankings $\sigma_1, \ldots, \sigma_t$ for any $1 \leq t < k$. These balls can each contain at most one ranking of an optimal solution, as otherwise two rankings in the solution are of distance strictly less than $\texttt{OPT}^{\alpha}_{\text{min}}$ by the triangle inequality. As there are fewer than k balls, they cannot cover all rankings in an optimal solution. Therefore, at every iteration there is a feasible ranking whose distance from all current centers is at least $\texttt{OPT}^{\alpha}_{\text{min}}/2$, and applying the oracle of Lemma 5.2 finds such a ranking. By a union bound, the probability that any of the $k-1$ oracle runs fails is at most $\zeta/(k-1)$. □

M Other Experimental Results

Here, we discuss the results of some additional experiments to answer RQ3 and RQ4.

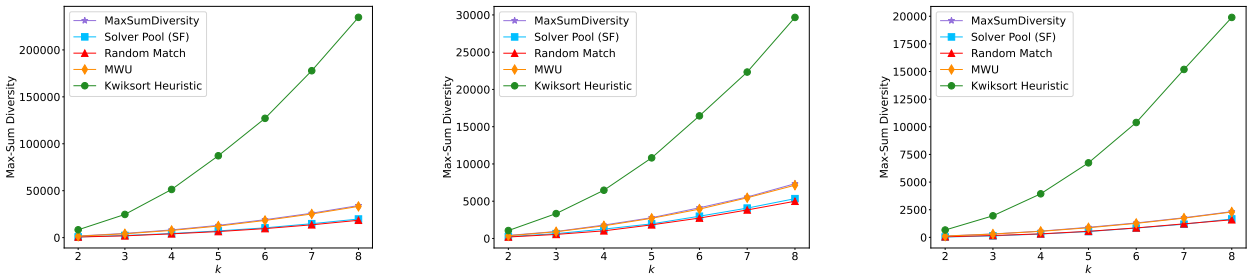
RQ3 - scalability. We study and plot the running time of the algorithms in figures 7a, 7b and 7c. We observe that all our implemented algorithms run extremely quickly except MWU, which takes up to 80 seconds on the MovieLens dataset, which has the largest number of candidates. This suggests our algorithm remains practical despite the high theoretical running time we have proven. Although we do not plot it, we observe that the CP-SAT exact solver implementation is extremely slow. It times out at 2 hours even for $k \geq 3$ for max-min, and $k \geq 4$ for max-sum on the university and country dataset, and for all $k \geq 2$ on the larger movielens dataset for both diversity problems. Therefore, our algorithms present much more scalable and practical solutions to this problem.

RQ4 - Robustness to Kendall tau distance. Finally, we turn to the question of the performance of our algorithms when studying diverse rank aggregation under the Kendall tau distance; that is, diversity of rankings is now computed based off the Kendall tau distance. We plot the results for the max-sum diverse rank aggregation in figures 8a, 8b and 8c, and the results for max-min diverse rank aggregation in figures 9a, 9b and 9c.



(a) Movielens dataset (b) University dataset (2015) (c) Country dataset (2014)

Figure 7: Running time of implemented algorithms



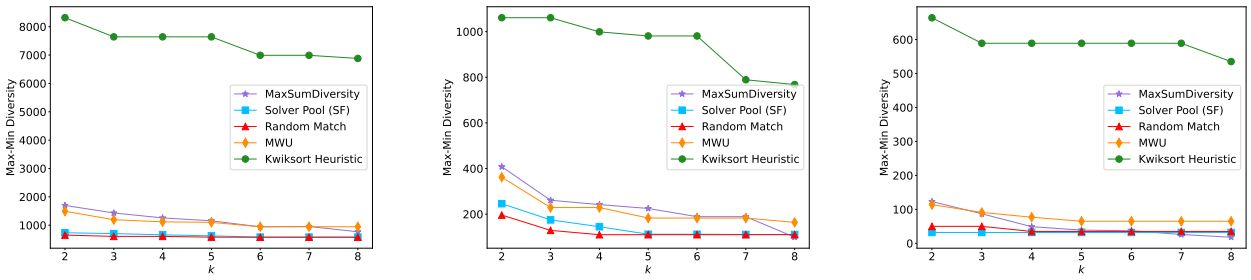
(a) Movielens dataset (b) University dataset (2015) (c) Country dataset (2014)

Figure 8: Performance of algorithms on max-sum diverse rank aggregation under *Kendall tau* distance

We observe that the performance of the algorithms for the max-sum and max-min diverse rank aggregation closely follows that observed for the Spearman footrule distance for both the max-sum and max-min objectives. In particular, MAXSUMDIVERSITY still produces output rankings with high diversity, and MWU tends to produce output rankings with good max-min diversity. We observe that KWIKSORT HEURISTIC performs far better than our algorithms at finding max-sum and max-min diverse rankings under Kendall tau distance, with significantly higher diversity quality than our algorithms. This suggests that the space of optimal Spearman footrule aggregate rankings is too constrained for finding diverse approximate rankings under Kendall tau distance.

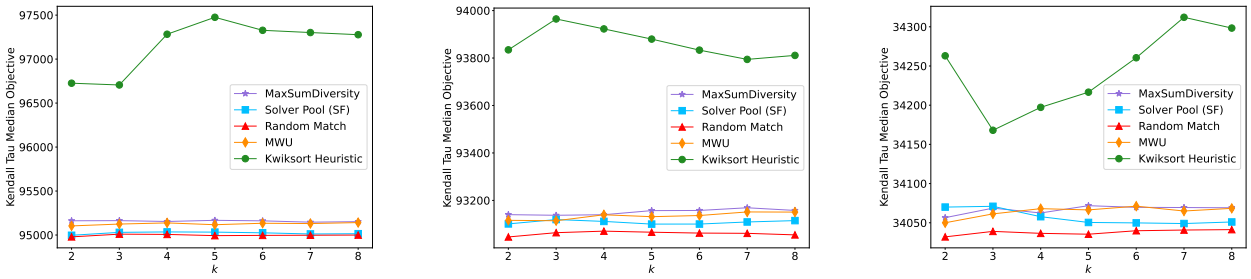
Recall that in the diverse rank aggregation problem, our algorithms must output rankings that are optimal aggregate rankings under Spearman footrule distance. However, the output rankings are not guaranteed to be optimal aggregate rankings under the Kendall tau distance; in fact, the only theoretical guarantee is that they are a 2-approximation of optimal (Diaconis & Graham, 1977). Further, KWIKSORT HEURISTIC is only a 11/7-approximation for rank aggregation under Kendall tau distance in expectation. We thus compute the rank-aggregation objective value under the Kendall tau distance for each ranking output by the algorithms, and plot the means in figures 10a, 10b, and 10c. Surprisingly, although theory predicts that Kwiksort outputs will produce a $11/7$-approximation and outperform our algorithms, which output a 2-approximation, we see the opposite. The Kendall tau rank-aggregation objective is much larger for KWIKSORT HEURISTIC than for all other baselines. This also offers a possible explanation for the diversity of KWIKSORT HEURISTIC, as it can output rankings from a much larger set of options and thus find more diverse rankings.

Therefore, unlike standard rank aggregation, where Spearman footrule algorithms can give good solutions in practice, the (approximate) diverse rank aggregation task under the Kendall tau distance may require more tailored algorithms.



(a) Movielens dataset (b) University dataset (2015) (c) Country dataset (2014)

Figure 9: Performance of algorithms on max-min diverse rank aggregation under *Kendall tau* distance



(a) Movielens dataset (b) University dataset (2015) (c) Country Dataset (2014)

Figure 10: Mean *Kendall tau* rank aggregation objective of rankings output by algorithms