On the Hardness of Maximum Rank Aggregation Problems☆

Christian Bachmaier[a], Franz J. Brandenburg[a,*], Andreas Gleißner[a], Andreas Hofmeier[b]

[a] University of Passau, 94030 Passau, Germany
[b] DocuWare GmbH, 82110 Germering, Germany

Abstract

The rank aggregation problem consists in finding a consensus ranking on a set of alternatives, based on the preferences of individual voters. The alternatives are expressed by permutations, whose pairwise distance can be measured in many ways.

In this work we study a collection of distances, including the Kendall tau, Spearman footrule, Minkowski, Cayley, Hamming, Ulam, and related edit distances. Unlike the common median by summation, we compute the consensus against the maximum. The maximum consensus attempts to minimize the discrimination against any voter and is a smallest enclosing ball or center problem.

We provide a general schema via local permutations for the **NP**-hardness of the maximum rank aggregation problems under all distances which satisfy some general requirements. This unifies former **NP**-hardness results for some distances and lays the ground for further ones. In particular, we establish a dichotomy for rank aggregation problems under the Spearman footrule and Minkowski distances: The median version is solvable in polynomial time whereas the maximum version is **NP**-hard. Moreover, we show that the maximum rank aggregation problem is 2-approximable under any pseudometric and fixed-parameter tractable under the Kendall tau, Hamming, and Minkowski distances, where again a general schema via modification sets applies.

Keywords: center consensus, rank aggregation, maximum ranking, permutation metrics, computational complexity, FPT

1. Introduction

The task of ranking a list of alternatives is encountered in many situations. A major goal is to find the best consensus. This task is known as the rank aggregation problem, and was widely studied in recent years [1–9]. The problem has numerous applications in sports, voting systems for elections, search engines, and evaluation systems on the web [7].

From mathematical and computational perspectives, the rank aggregation problem is given by a set of m permutations on a set of size n, and the goal is to find a consensus permutation with minimum distance to the given permutations. There are many ways to measure the distance between two permutations and to aggregate the cost by an objective function. Various distances are based on primitive operations on permutations, as they are used in sorting algorithms and string matching. Aggregation is by taking the sum or the maximum.

For the rank aggregation problem Kemeny, [10] proposed to count the pairwise disagreements between the orderings of two items, which is commonly known as the Kendall tau distance. For permutations it is the "bubble sort" distance, i. e., the number of pairwise adjacent transpositions needed to transform one permutation into the other, or the number of crossings in a two-layered drawing of a bipartite graph with vertices of 1 to n on each layer and edges $\{i, i\}$ for $i = 1, \ldots, n$ [11]. Another popular measure is the Spearman footrule distance [12], which is the L_1-norm of two n-dimensional vectors and expresses the total movement of items.

The geometric median of the input permutations is commonly taken for the aggregation, which means *summing* up the cost of comparing each input permutation with the consensus. From the computational perspective this makes an essential difference between the Spearman footrule and the Kendall tau distances, since the further allows a polynomial time solution via weighted bipartite matching [7], whereas the latter leads to an **NP**-hard rank aggregation problem [6], even for four voters [7, 11]. It has a PTAS [8]

☆Supported by the Deutsche Forschungsgemeinschaft (DFG), grant Br835/16-1.
*Corresponding author.

Email addresses: bachmaier@fim.uni-passau.de (Christian Bachmaier), brandenb@informatik.uni-passau.de (Franz J. Brandenburg), gleissner@fim.uni-passau.de (Andreas Gleißner), andreas.hofmeier@docuware.com (Andreas Hofmeier)

and is fixed-parameter tractable [3, 9].

Here we study the *maximum version*, which is also known as a smallest enclosing ball or center problem. The aim is to avoid a discrimination of a single voter or permutation against the consensus. The objective is a minimum k such that all permutations are within distance k from the consensus. Biedl et al. [11] studied this version for the Kendall tau distance and showed that it is **NP**-hard to determine whether there is a permutation τ which is within a distance of at most k to all input permutations, even for any $m \geq 4$ permutations. The **NP**-hardness was independently proven by Popov [13] and further investigated by Schwarz [14]. The smallest enclosing ball problem is a famous mathematical problem. It dates back to Sylvester in 1857 [15] and has been intensively studied in computational geometry [16], production planning [17], and stringology [18].

Besides the Kendall tau and the Spearman footrule distances there are other distance measures on permutations [7, 10, 19]. Many of them are edit distances, which can be expressed as the minimum number of specific primitive operations to transform one permutation into the other. Some operations are local, others operate globally on singletons, and the most powerful ones manipulate blocks or subsequences in a single step. The swap of two adjacent items, a unit movement of an item and a substitution are local operations and are used for the Kendall tau, Spearman footrule, and Hamming distances, respectively, whereas the Cayley distance allows the exchange of two items at arbitrary positions. The block reversal distance counts the reversal of a block in a permutation as a unit step. In consequence, the distance between two permutations often varies by a factor of $\mathcal{O}(n)$, e. g., if the first and last candidates are interchanged or if the second is the reversal of the first permutation. Such permutations are within unit distance for the block reversal distance and $\mathcal{O}(n^2)$ for the Kendall tau and Spearman footrule distances. As shown by Diaconis and Graham [12], these two distances are within a factor of two. The same applies to the Hamming and Cayley distances. Thus, these pairs meet the *metric boundedness property* [20]. For a broad discussion of distances we refer to [19]. Since computing the block reversal or the block transposition distance is **NP**-hard [21, 22], we do not expect that maximum ranking under these distances is efficiently solvable and refrain from treating them any further.

We extend the collection of distances on permutations by Swap-and-Mismatch, Damerau-Levenshtein, and Lee distances, which are used in combinatorics for genome comparisons [19]. Our main contribution is a general schema for the complexity analysis of maximum rank aggregation problems, which allows us to prove **NP**-hardness and fixed-parameter tractability under any metric which satisfies some requirements. These requirements are met by our collection of distances. We associate the maximum rank aggregations on permutations and the string consensus problem on strings. Permutations on a set of size n can be seen as strings on an alphabet of size n, where each element occurs exactly once. However, the alphabet must scale with the length of the permutation and the uniqueness of the elements makes them special as strings.

For the association we use the generalization of total to bucket orders and local permutations as extensions of bucket orders. The technique of local permutations was first used implicitly by Popov [13] for Kendall tau and Cayley distances and with the main focus on the string consensus problem. Thereafter we obtain the **NP**-hardness results by reductions from the Closest Binary String and Hitting String problems, which is more general than the previous reductions [6, 7, 11, 13].

The paper is organized as follows. After some preliminaries in Sect. 2 we show in Sect. 3 that Maximum Ranking (MR) is tractable under the Maximum distance, whereas MR is intractable under many other distances as shown in Sect. 4. In Sect. 5 we establish that MR is 2-approximable for pseudometrics. Finally, in Sect. 6, we present fixed-parameter algorithms to solve MR under various distances.

In a preliminary version of this paper [23] presented at IWOCA 2013 we consider only a subset of the distances, but our generalized schema applies to a broader set.

2. Preliminaries

For a binary relation ρ on a domain $\mathcal{D}$ and for each $x, y \in \mathcal{D}$, we write $x <_\rho y$ if $(x, y) \in \rho$ and $x \not<_\rho y$ if $(x, y) \notin \rho$. A binary relation κ is a (strict) *partial order* if it is *irreflexive*, *asymmetric* and *transitive*, i. e., $x \not<_\kappa x$, $x <_\kappa y \Rightarrow y \not<_\kappa x$, and $x <_\kappa y \wedge y <_\kappa z \Rightarrow x <_\kappa z$ for all $x, y, z \in \mathcal{D}$. Candidates x and y are called *unrelated by* κ if $x \not<_\kappa y \wedge y \not<_\kappa x$, which we denote by $x \ngtrless_\kappa y$. The intuition of $x <_\kappa y$ is that κ *ranks* x *before* y, which means a preference for x. If $x <_\kappa y$ or $y <_\kappa x$, we speak of a *constraint of* κ *on* x *and* y. For $\mathcal{X}, \mathcal{Y} \subseteq \mathcal{D}$ we denote $\mathcal{X} <_\kappa \mathcal{Y}$ if $x <_\kappa y$ for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$, and define $x <_\kappa \mathcal{Y}$ and $\mathcal{X} <_\kappa y$ accordingly. The intersection of two partial orders $\mu \cap \kappa$ is a partial order consisting of all pairs of candidates where μ and κ agree.

A *total order* is a *complete* partial order, i. e., $x <_\tau y \vee y <_\tau x$ for all $x, y \in \mathcal{D}$ with $x \neq y$. Let $n = |\mathcal{D}|$. For every total order τ there is a unique *permutation*, i. e., a bijection $\tau' : \mathcal{D} \to \{1, \ldots, n\}$ such that $x <_\tau y \Leftrightarrow \tau'(x) < \tau'(y)$. In the rest of the paper we identify total orders and their corresponding permutations, taking the view whichever comes in more handy. The set of permutations on $\mathcal{D}$ is denoted by $\mathrm{Perm}(\mathcal{D})$. We denote the permutation $\{x_1, \ldots, x_n\} \to \{1, \ldots, n\} : x_i \mapsto i$ by $[x_1 x_2 \ldots x_n]$.

We distinguish between permutations in $\mathrm{Perm}(\mathcal{D})$ representing votes on an arbitrary candidate set $\mathcal{D}$ and permutations in $\mathrm{Perm}(\{1, \ldots, n\})$ representing an exchange of positions, i. e., transformations on votes. Let $\tau \in \mathrm{Perm}(\mathcal{D})$ be a vote, $T \in \mathrm{Perm}(\{1, \ldots, n\})$ be an exchange of positions and $\tau' = T \circ \tau$. Then $\tau' \in \mathrm{Perm}(\mathcal{D})$, i. e., τ' can be seen as another vote obtained from τ by applying a change represented by T.

A *transposition* is a permutation on $\{1, \ldots, n\}$ switching the positions of two candidates. Hence, for positions $i, j \in \{1, \ldots, n\}$, we define the transposition $T_{i,j} \in \mathrm{Perm}(\{1, \ldots, n\})$ by $T_{i,j}(i) = j$, $T_{i,j}(j) = i$ and $T_{i,j}(k) = k$ for $k \notin \{i, j\}$. Transpositions can also be considered as operations acting on permutations on $\mathcal{D}$. For $x, y \in \mathcal{D}$ and $\sigma \in \mathrm{Perm}(\mathcal{D})$ we call $T_{\sigma(x),\sigma(y)} \circ \sigma \in \mathrm{Perm}(\mathcal{D})$ the *transposition of x and y in σ*. Transpositions $T_{i,j}$ of adjacent candidates with $|i - j| = 1$ are called *swaps*.

For the connection between consensus problems on strings and permutations we use bucket orders and their extensions. A total order $\tau \in \mathrm{Perm}(\mathcal{D})$ is a *total extension* of a partial order κ if τ does not contradict κ, i. e., $x <_\kappa y$ implies $x <_\tau y$ for all $x, y \in \mathcal{D}$. We denote the set of total extensions of a partial order κ by $\mathrm{Ext}(\kappa)$.

A *bucket order* is a partial order κ for which unrelatedness $\ngtrless_\kappa$ is transitive. Then $\ngtrless_\kappa$ is an equivalence relation whose equivalence classes are called *buckets*. In other words, κ induces a total order order on the buckets while candidates of the same bucket are unrelated, see [1, 2, 24]. An extension of a bucket order preserves the total order of the buckets and allows any permutation of the candidates within each bucket.

A binary function $d : \mathrm{Perm}(\mathcal{D}) \times \mathrm{Perm}(\mathcal{D}) \to \mathbb{R}$ is called a *pseudometric* if $d(\sigma, \tau) \geq 0$, $d(\sigma, \tau) = d(\tau, \sigma)$, $\sigma = \tau \Rightarrow d(\sigma, \tau) = 0$, and $d(\sigma, \tau) + d(\tau, \rho) \geq d(\sigma, \rho)$ for all $\sigma, \tau, \rho \in \mathrm{Perm}(\mathcal{D})$. It is a *metric* if, additionally, $d(\sigma, \tau) = 0$ implies $\sigma = \tau$.

A *string* s of length k over an alphabet $\mathcal{D}$ is a k-tuple in $\mathcal{D}^k$ or a mapping $s : \{1, \ldots, k\} \to \mathcal{D}$ from positions to characters. We use both notions interchangeably. We say that a string s *represents* a permutation ϕ if s is bijective and $s^{-1} = \phi$. Then the alphabet $\mathcal{D}$ is taken as a candidate set and vice versa.

Next we introduce the main concept of this work: The maximum version of the rank aggregation problem under various distances [19, 25, 26].

Definition 1 (Maximum Ranking (MR))**.**
Instance: A set $\mathcal{D}$ of n candidates, a list of m voters $\sigma_1, \ldots, \sigma_m \in \mathrm{Perm}(\mathcal{D})$, and an integer $k \in \mathbb{N}$.
*Question: Does there exist a permutation $\tau \in \mathcal{D}$ (called k-*consensus*) with $\max_{j=1}^{m} d(\sigma_j, \tau) \leq k$?*

The maximum ranking problem is also known as the smallest enclosing ball or center problem [14, 15]. The k-consensus τ guarantees a distance of at most k to the preferences of all voters and avoids the discrimination of any voter. The maximum ranking problem is investigated under several distances, which evaluate disagreements differently. They are defined next.

Let σ and $\tau \in \mathrm{Perm}(\mathcal{D})$ be two permutations. Define the set of *dirty* pairs $\mathcal{K}(\sigma, \tau) = \{\{x, y\} \subseteq \mathcal{D} : x <_\sigma y \wedge y <_\tau x\}$ as the set of pairs of candidates $x, y \in \mathcal{D}$ where σ and τ disagree on their order. Then the *Kendall tau distance* K is defined by $K(\sigma, \tau) = |\mathcal{K}(\sigma, \tau)|$. It coincides with the minimum number k of swaps $T_1, \ldots, T_k$ such that $\tau = T_k \circ \ldots \circ T_1 \circ \sigma$. We obtain the *Cayley distance* $C(\sigma, \tau)$ if additionally non-adjacent candidates can be exchanged. $C(\sigma, \tau)$ is the minimum number of transpositions $T_1, \ldots, T_k$ such that $\tau = T_k \circ \ldots \circ T_1 \circ \sigma$. The Cayley distance can also be expressed as $C(\sigma, \tau) = n - \sharp\mathcal{C}(\tau \circ \sigma^{-1})$ [26], where $\sharp\mathcal{C}(\rho)$ is the number of cycles of a permutation ρ. A cycle $\mathcal{C} = (x_1 x_2 \ldots x_{|\mathcal{C}|})$ of $\rho \in \mathrm{Perm}(\{1, \ldots, n\})$ is a (cyclic) sequence of distinct candidates such that $\rho(x_i) = x_{i+1}$ for $1 \leq i < |\mathcal{C}|$ and $\rho(x_{|\mathcal{C}|}) = x_1$. The cycles form a partition of $\{1, \ldots, n\}$ and can be used to specify any permutation.

Define the set of *displaced* candidates by $\mathcal{H}(\sigma, \tau) = \{x \in \mathcal{D} : \sigma(x) \neq \tau(x)\}$ as the set of candidates $x \in \mathcal{D}$ where σ and τ disagree on their position. The *Hamming distance* H is defined by $H(\sigma, \tau) = |\mathcal{H}(\sigma, \tau)|$, which is the number of positions $i \in \{1, \ldots, n\}$ where $\sigma^{-1}(i) \neq \tau^{-1}(i)$. This view is also taken by the Hamming distance between binary strings $s, t \in \{0, 1\}^n$ defined by $H(s, t) = |\{i \in \{1, \ldots, n\} : s(i) \neq t(i)\}|$, where $s(i)$ is the i-th character of s.

Edit distances take the minimum number of operations from some predefined set to change one string into an other. For example, the Hamming

	Insert/Delete	Substitute	Swap
Kendall tau K			$\times$
Hamming H		$\times$	
Swap-and-Mismatch S		$\times$	$\times$
Ulam U	$\times$	$\times$	
Damerau-Levenshtein D	$\times$	$\times$	$\times$

Table 1: Operations for some edit distances

distance is an edit distance for the operation of substituting single characters. For an overview see Table 1. In general, every edit distance on strings can also be seen as a permutation metric. Note that in the sequence of strings constituting the step-by-step transformation, only the first and the last string actually need to represent a permutation, whereas the immediate strings may also have duplicates or missing candidates.

The Levenshtein distance U is the edit distance where the operations are substitutions, insertions, or deletions of a single character [27]. In the context of permutations it is known as the *Ulam distance*. A different characterization is as follows. Let $\sigma, \tau \in \mathrm{Perm}(\mathcal{D})$ be two permutations. A tuple $(x_1, \ldots, x_l)$ with $x_i \in \mathcal{D}$ is a *common subsequence* of σ and τ if $i < j \Leftrightarrow x_i <_\sigma x_j \wedge x_i <_\tau x_j$. Let $\mathrm{lcs}(\sigma, \tau) = \max\{\, l \colon (x_1, \ldots, x_l)$ is a common subsequence of σ and $\tau \,\}$. Then the Ulam distance is $U(\sigma, \tau) = n - \mathrm{lcs}(\sigma, \tau)$.

The *Damerau-Levenshtein distance* D unites the operation sets of the Ulam and Kendall tau distances [28]. Another variant is the *Swap-and-Mismatch* edit distance, which only allows for swaps and substitutions [29].

Next we turn to metrics which are based on summing up positional differences rather than edit operations.

Define the *Minkowski distance* by $F_p(\sigma, \tau) = \left(\sum_{x \in \mathcal{D}} |\sigma(x) - \tau(x)|^p\right)^{\frac{1}{p}}$ for $p \in \mathbb{N} \setminus \{0\}$. F_1 is also known as the *Spearman footrule distance* or taxicab metric. F_2 is the Euclidean metric and also known as the Spearman rho distance [26]. To simplify proofs we introduce the notion of the *raised Minkowski distance* $\hat{F}_p$ defined by $\hat{F}_p(\tau, \sigma) = (F_p(\tau, \sigma))^p = \sum_{x \in \mathcal{D}} |\tau(x) - \sigma(x)|^p$.

One can also consider the limit for $p \to \infty$ and $p \to -\infty$. The Chebyshev or *Maximum distance* is $F_\infty(\sigma, \tau) = \max_{x \in \mathcal{D}} |\sigma(x) - \tau(x)|$. Define the *Minimum distance* by $F_{-\infty}(\sigma, \tau) = \min_{x \in \mathcal{D}} |\sigma(x) - \tau(x)|$. Note that $F_{-\infty}$ is

	Complexity	FPT for distance k
Kendall tau K	**NP**-complete (also in [11, 14])	$\mathcal{O}((2k)^k \cdot (mn \log n + k))$
Cayley C	**NP**-complete	
Hamming H	**NP**-complete	$\mathcal{O}((2k)^k \cdot mn)$
Ulam U	**NP**-complete	
Damerau-Levenshtein D	**NP**-complete	
Swap-and-Mismatch S	**NP**-complete	
Lee L	**NP**-complete	
Minkowski F_p, $p \in \mathbb{N} \setminus \{0\}$	**NP**-complete	$\mathcal{O}((2k^{p+1})^{\lceil \frac{k^p}{p+1} \rceil} \cdot mn)$
Spearman footrule F_1	**NP**-complete	$\mathcal{O}((2k)^k \cdot mn)$
Minimum $F_{-\infty}$	**NP**-complete	fixed-parameter intractable
Maximum F_∞	$\mathcal{O}(n\,(\log n + m))$	

Table 2: Summary of complexity results for the maximum ranking problem

not a metric and satisfies only non-negativity and symmetry.

Finally, the *Lee distance* is $L(\sigma, \tau) = \sum_{x \in \mathcal{D}} \min\{|\sigma(x) - \tau(x)|, n - |\sigma(x) - \tau(x)|\}$ [30]. Roughly speaking, it may be regarded as a variant of the Spearman footrule distance F_1 where the positions are arranged in a circular manner rather than linear such that the first and the last position are taken as next to each other.

Our complexity results on the maximum rank aggregation problem under diverse distances are summarized in Table 2.

3. Efficient Algorithms

First, we consider a case where MR is efficiently solvable.

Theorem 1. *MR is efficiently solvable under the Maximum distance F_∞.*

Proof. To find a permutation τ satisfying $\max_{j=1}^{m} \max_{x \in \mathcal{D}} |\sigma_j(x) - \tau(x)| \leq k$, we solve a maximum matching problem in the bipartite graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with vertices $\mathcal{V} = \mathcal{D} \cup \{1, \ldots, n\}$ and an edge $(x, i) \in \mathcal{E}$ if $\max_{j=1}^{m} |\sigma_j(x) - i| \leq k$. Every matching of size n corresponds to a k-consensus τ and vice versa. As $|\mathcal{E}| < n(2k + 1)$, this can be done in $\mathcal{O}(n^2 \cdot k)$ time. For an improvement observe that the suitable positions for each candidate are consecutive, thus form an interval. To each candidate $x \in \mathcal{D}$ assign the interval $I_x = \{i \in \{1, \dots, n\} : \max_{j=1}^{m} |\sigma_j(x) - i| \leq k\}$. Then, iterate over the positions $i \in \{1, \dots, n\}$. In step i, select the candidate to place at position i. Choose from those candidates x with $i \in I_x$ and which have not been placed before. If there are multiple suitable candidates, prefer a candidate whose interval has the least upper endpoint. In the case that there are no suitable candidates, reject the instance. We use a heap to manage the intervals of unplaced candidates, inserting the interval once we reach its lower endpoint. Determining the endpoints of the intervals can be done in $\mathcal{O}(n \cdot m)$ and the iteration is done in $\mathcal{O}(n \log n)$, resulting in a total running time of $\mathcal{O}(n(\log n + m))$. $\square$

4. Intractability Results

We prove that MR is **NP**-complete under the Hamming, Minkowski, Kendall tau, Cayley, Ulam, Damerau-Levenshtein, Swap-and-Mismatch, Lee and Minimum distances. As these distances can be efficiently computed for two total orders [11, 25, 29, 31, 32], membership is in **NP**.

For the **NP**-hardness proofs we develop a general schema, which generalizes techniques from [13]. First, we prove that the **NP**-complete CLOSEST BINARY STRING problem [33] can be reduced to a special case of MR under any metric subject to Requirements 1 and 2 defined below. Then, we show that these requirements are satisfied by the aforementioned metrics except the Minimum distance, for which we provide a reduction from the **NP**-complete HITTING STRING problem [34].

Definition 2 (CLOSEST BINARY STRING [33])**.**
Instance: $k, n \in \mathbb{N}$*, a list* $s_1, \dots, s_m \in \{0,1\}^n$ *of* m *binary strings of length* n*.*
Question: Does there exist a string $t \in \{0,1\}^n$ *with* $\max_{j=1}^{m} H(s_j, t) \leq k$*?*

For the rest of this section, let $\mathcal{D}$ be a set of $2n$ candidates. We arbitrarily partition $\mathcal{D}$ into n disjoint 2-element sets $\mathcal{B}_i = \{a_i, b_i\}$ called *buckets*.

Definition 3 (local permutation)**.**
A permutation is local if it is an extension of a bucket order κ *on* $\mathcal{D}$ *with buckets* $\mathcal{B}_i$ *ordered by* $\mathcal{B}_1 <_\kappa \dots <_\kappa \mathcal{B}_n$

We state the following properties to be met by a metric d in order to be applicable in the forthcoming reduction.

Requirement 1 (optimality of local permutations)**.** *Let* $\sigma_1, \ldots, \sigma_m \in \mathrm{Ext}(\kappa)$ *be local permutations and* $k \in \mathbb{N}$*. If there is a* k*-consensus* $\tau \in \mathrm{Perm}(\mathcal{D})$ *with* $\max_{j=1}^{m} d(\sigma_j, \tau) \leq k$*, then there is a local permutation* $\tau' \in \mathrm{Ext}(\kappa)$ *with* $\max_{j=1}^{m} d(\sigma_j, \tau') \leq k$*.*

In other words, if all voters are local and our metric meets Requirement 1, then we can safely demand that the consensus is local, too, without impairing its chance to satisfy the upper bound k. Note that distance d satisfies Requirement 1 if for every local permutation $\sigma \in \mathrm{Ext}(\kappa)$ and permutation $\tau \in \mathrm{Perm}(\mathcal{D})$ we can find a $\tau' \in \mathrm{Ext}(\kappa)$ such that $d(\sigma, \tau') \leq d(\sigma, \tau)$.

The second requirement puts tight constraints on the distance of local permutations.

Requirement 2 (distance constraints)**.** *There is a constant* $c > 0$ *such that for all local permutations* $\sigma, \tau \in \mathrm{Ext}(\kappa)$ *the distance is* $d(\sigma, \tau) = c \cdot |\mathcal{K}(\sigma, \tau)|$*.*

Requirement 2 tightens the *metric boundedness property* [20] when restricted to local permutations. Two metrics d and d' are related by the metric boundedness property if there are constants c_1 and c_2 such that $c_1 d'(\sigma, \tau) \leq d(\sigma, \tau) \leq c_2 d'(\sigma, \tau)$ for arbitrary permutations $\sigma, \tau \in \mathrm{Perm}(\mathcal{D})$. If d and d' satisfy Requirement 2 for constants c and c', respectively, they are tied by $d'(\sigma, \tau) = \frac{c'}{c} d(\sigma, \tau)$ for local permutations $\sigma, \tau \in \mathrm{Ext}(\kappa)$.

Note that all local permutations agree on the order of candidates from different buckets. Thus, a distance satisfying Requirement 2 is exactly a constant multiple of the number of buckets $\mathcal{B}_i$ where one permutation ranks a_i before b_i and the other ranks b_i before a_i.

Now we can state our schema for the **NP**-hardness reduction.

Theorem 2. *MR under a metric* d *is* **NP***-hard if* d *satisfies Requirements 1 and 2.*

Proof. Consider an instance of Closest Binary String consisting in a list $s_1, \ldots, s_m \in \{0,1\}^n$ of m binary strings of length n and an upper bound $k \in \mathbb{N}$ as in Definition 2. We choose the candidate set $\mathcal{D} = \bigcup_{i=1}^{n} \mathcal{B}_i$. Consider the bijective mapping $f : \{0,1\}^n \to \mathrm{Ext}(\kappa)$, which encodes a string s of length n as a local permutation where $a_i <_{f(s)} b_i$ if $s(i) = 0$ and $b_i <_{f(s)} a_i$ if $s(i) = 1$. More formally, $f(s)(a_i) = 2i - 1 + s(i)$ and $f(s)(b_i) = 2i - s(i)$ for all strings $s \in \{0,1\}^n$. For instance, $f(\texttt{"010"}) = [\underbrace{a_1 b_1}_{\mathcal{B}_1} \underbrace{b_2 a_2}_{\mathcal{B}_2} \underbrace{a_3 b_3}_{\mathcal{B}_3}]$, see also [13]. Observe that for strings $s, t \in \{0,1\}^n$ and $i \in \{1, \ldots, n\}$ we have $s(i) \neq t(i)$ if and only if $\{a_i, b_i\} \in \mathcal{K}(f(s), f(t))$. For each string s_j introduce the voter $\sigma_j = f(s_j)$ and let $k' = c \cdot k$, where c is the constant from Requirement 2.

Suppose that a string $t^* \in \{0,1\}^n$ satisfies $\max_{j=1}^m H(s_j, t^*) \leq k$. Let $j \in \{1, \ldots, m\}$. We have

$$\begin{aligned} k' = c \cdot k \geq c \cdot H(s_j, t^*) &= c \cdot |\{i \in \{1, \ldots, n\} : s_j(i) \neq t^*(i)\}| \\ &= c \cdot |\mathcal{K}(f(s_j), f(t^*))| = d(\sigma_j, f(t^*)) \end{aligned}$$

by Requirement 2 and the fact that $f(s_j)$ and $f(t)$ are local permutations. Therefore, $f(t^*)$ is a k'-consensus for the MR problem.

Conversely, suppose that τ^* satisfies $\max_{j=1}^m d(\sigma_j, \tau^*) \leq k'$. W. l. o. g. assume that τ^* is local by Requirement 1. Again, let $j \in \{1, \ldots, m\}$. By Requirement 2 we obtain

$$\begin{aligned} k = \frac{k'}{c} \geq \frac{1}{c} \cdot d(\sigma_j, \tau^*) = |\mathcal{K}(\sigma_j, \tau^*)| &= \left|\{i \in \{1, \ldots, n\} : f^{-1}(\sigma_j) \neq f^{-1}(\tau^*)\}\right| \\ = H(s_j, f^{-1}(\tau^*))\,, \end{aligned}$$

i. e., the string $t^* = f^{-1}(\tau^*) \in \{0,1\}^n$ satisfies $\max_{j=1}^m H(s_j, t^*) \leq k$.

We conclude that there is a binary string $t^* \in \{0,1\}^n$ with $\max_{j \in \{1,\ldots,m\}} H(s_j, t^*) \leq k$ if and only if there is a permutation $\tau^* \in \mathrm{Perm}(\mathcal{D})$ with $\max_{j \in \{1,\ldots,m\}} d(f(s_j), \tau^*) \leq c \cdot k$. □

Next we show that Requirements 1 and 2 are met by the Kendall tau, Cayley, Hamming, Ulam, Damerau-Levenshtein, Swap-and-Mismatch, Lee, and Minkowski distances.

Lemma 1. *Let $\{x, y\} \in \mathcal{K}(\sigma, \tau)$ be a dirty pair of candidates where two permutations σ and $\tau \in \mathrm{Perm}(\mathcal{D})$ disagree on their order. Then the Kendall tau distance strictly decreases if we transpose x and y in τ, i. e., $K(\sigma, T_{\tau(x),\tau(y)} \circ \tau) < K(\sigma, \tau)$.*

Proof. Let $\tau' = T_{\tau(x),\tau(y)} \circ \tau$. W. l. o. g. assume $x <_\tau y$. Now compare the set $\mathcal{K}^+ = \mathcal{K}(\sigma, \tau') \setminus \mathcal{K}(\sigma, \tau)$ with the set $\mathcal{K}^- = \mathcal{K}(\sigma, \tau) \setminus \mathcal{K}(\sigma, \tau')$. Then $K(\sigma, \tau') < K(\sigma, \tau)$ if $|\mathcal{K}^+| < |\mathcal{K}^-|$. Let $Z_<, Z_|$ and $Z_>$ be the candidates that are ranked by σ before, between, and after x and y, respectively. Formally, $Z_< = \{z \in \mathcal{D} : x <_\tau z <_\tau y \wedge z <_\sigma y <_\sigma x\}$, $Z_| = \{z \in \mathcal{D} : x <_\tau z <_\tau y \wedge y <_\sigma z <_\sigma x\}$, and $Z_> = \{z \in \mathcal{D} : x <_\tau z <_\tau y \wedge y <_\sigma x <_\sigma z\}$. By a distinction of cases we obtain

$$\mathcal{K}^+ = \bigcup_{z \in Z_<} \{\{y,z\}\} \cup \bigcup_{z \in Z_>} \{\{x,z\}\}\,, \text{ and}$$
$$\mathcal{K}^- = \bigcup_{z \in Z_<} \{\{x,z\}\} \cup \bigcup_{z \in Z_>} \{\{y,z\}\} \cup \bigcup_{z \in Z_|} \{\{x,z\},\{y,z\}\} \cup \{\{x,y\}\}\,.$$

Hence, $K(\sigma,\tau') = K(\sigma,\tau) - |Z_||- 1$. □

Lemma 2. *Let τ^* be an optimal consensus for the MR problem under the Kendall tau distance K with voters $\sigma_1,\ldots,\sigma_m$. Let $\mu = \bigcap_{j=1}^m \sigma_j$ be the partial order with $x <_\mu y$ if and only if $x <_{\sigma_j} y$ for all $j \in \{1,\ldots,m\}$. Then $\tau^* \in \mathrm{Ext}(\mu)$.*

Proof. Assume by contradiction that there are candidates $x, y \in \mathcal{D}$ with $x <_\mu y$ and $y <_{\tau^*} x$. Then $x <_{\sigma_j} y$ and $\{x,y\} \in \mathcal{K}(\sigma_j,\tau^*)$ for every $j \in \{1,\ldots,m\}$. Thus, $\max_{j=1}^m d(\sigma_j, T_{\tau^*(x),\tau^*(y)} \circ \tau^*) < \max_{j=1}^m d(\sigma_j,\tau^*)$ by Lemma 1, which is a contradiction to the optimality of τ^*. □

Corollary 1. *The Kendall tau distance K satisfies Requirements 1 and 2.*

Proof. Let $\sigma_1,\ldots,\sigma_m \in \mathrm{Ext}(\kappa)$ be local permutations and $\mu = \bigcap_{j=1}^m \sigma_j$. Every extension of μ is also an extension of κ, since $\kappa \subseteq \mu$ is a binary relation. Hence, Requirement 1 follows immediately from Lemma 2. Let $c = 1$. Then Requirement 2 is just the definition of the Kendall tau distance restricted to local permutations. □

Note that our general schema shows the **NP**-hardness of MR under the Kendall tau distance for many voters, which parallels the result from [13]. However, the problem is known to be **NP**-hard even for four voters [7, 11].

For the proof of the following lemmas, we define two operations transforming an arbitrary into a local permutation.

For a permutation $\tau \in \mathrm{Perm}(\mathcal{D})$ let $\mathcal{A}_\tau = \{x \in \mathcal{D} : x \in \mathcal{B}_i \text{ and } \tau(x) \notin \{2i-1, 2i\}\}$ be the set of candidates which are not placed by τ in positions belonging to their bucket and let $I_\tau = \{\tau(x) : x \in \mathcal{A}_\tau\}$ be the set of positions taken by those candidates. Define the *position preserving localization* $l_{\mathrm{pos}} : \mathrm{Perm}(\mathcal{D}) \to \mathrm{Ext}(\kappa)$ as follows. Let $l_{\mathrm{pos}}(\tau)(x) = \tau(x)$ if $x \notin \mathcal{A}_\tau$, i. e., all candidates placed on positions corresponding to their bucket are unaffected. The remaining candidates in $\mathcal{A}_\tau$ are reordered by l_{pos} and assigned to free positions in I_τ to obtain a local permutation. Define $l_{\mathrm{pos}}(\tau)|_{\mathcal{A}_\tau} : \mathcal{A}_\tau \to I_\tau$ such that $x <_\kappa y \Rightarrow x <_{l_{\text{pos}}(\tau)} y$ for $x, y \in \mathcal{A}_\tau$. As a tie breaker use $a_i <_{l_{\text{pos}}(\tau)} b_i$ if $\mathcal{B}_i = \{a_i, b_i\} \subseteq \mathcal{A}_\tau$.

The *order preserving localization* $l_{\text{ord}} : \text{Perm}(\mathcal{D}) \to \text{Ext}(\kappa)$ is derived from the refinements of partial orders introduced by Fagin [24]. The local permutation $l_{\text{ord}}(\tau)$ is defined by $x <_{l_{\text{ord}}(\tau)} y$ if and only if $x <_\kappa y \vee (x \ngtrless_\kappa y \wedge x <_\tau y)$ for all $x, y \in \mathcal{D}$.

Both localization operations yield local permutations. They differ in that l_{pos} changes the position of as few candidates as possible while l_{ord} preserves the order of candidates from the same bucket. For example, let $\tau = [a_2 b_1 b_2 a_1]$. Then $l_{\text{pos}}(\tau) = [a_1 b_1 b_2 a_2]$, since l_{pos} preserves the position of b_1 and b_2, and $l_{\text{ord}}(\tau) = [b_1 a_1 a_2 b_2]$, since l_{ord} preserves $b_1 < a_1$ and $a_2 < b_2$.

Lemma 3. *The Cayley distance C satisfies Requirement 2.*

Proof. Let $\sigma, \tau \in \text{Ext}(\kappa)$ be local permutations. Since σ and τ agree on the order of candidates in different buckets, $\mathcal{K}(\sigma, \tau) \subseteq \{\, \mathcal{B}_i \colon i \in \{1, \ldots, n\}\,\}$. Consider a bucket $\mathcal{B}_i = \{a_i, b_i\}$. If $\mathcal{B}_i \in \mathcal{K}(\sigma, \tau)$, then a_i and b_i form a single cycle $(a_i b_i)$ of size 2 in $\tau \circ \sigma^{-1}$ as $\sigma(a_i) = \tau(b_i)$ and $\sigma(b_i) = \tau(a_i)$. If otherwise $\mathcal{B}_i \notin \mathcal{K}(\sigma, \tau)$, a_i and b_i each form a cycle of size 1. Thus, $C(\sigma, \tau) = 2n - \sharp\mathcal{C}(\tau \circ \sigma^{-1}) = 2n - |\mathcal{K}(\sigma, \tau)| - 2 \cdot |\{\, \mathcal{B}_i \colon i \in \{1, \ldots, n\}\,\} \setminus \mathcal{K}(\sigma, \tau)| = |\mathcal{K}(\sigma, \tau)| = K(\sigma, \tau)$. □

Lemma 4. *The Cayley distance C satisfies Requirement 1. In particular, $C(l_{\text{pos}}(\tau), \sigma) \leq C(\tau, \sigma)$ for every local permutation $\sigma \in \text{Ext}(\kappa)$ and permutation $\tau \in \text{Perm}(\mathcal{D})$.*

Proof. For $x \in \mathcal{D}$, denote by $|\mathcal{C}_x(\tau \circ \sigma^{-1})|$ the size of the cycle in $\tau \circ \sigma^{-1}$ containing x. If $\sigma(x) \neq \tau(x)$, then $|\mathcal{C}_x(\tau \circ \sigma^{-1})| \geq 2$, but $|\mathcal{C}_x(l_{\text{pos}}(\tau) \circ \sigma^{-1})| \leq 2$ as shown in the proof of Lemma 3. If otherwise $\sigma(x) = \tau(x)$, then $|\mathcal{C}_x(\tau \circ \sigma^{-1})| = |\mathcal{C}_x(l_{\text{pos}}(\tau) \circ \sigma^{-1})| = 1$. Observe that $\sum_{x \in \mathcal{D}} \frac{1}{|\mathcal{C}_x(\tau \circ \sigma^{-1})|} = \sharp\mathcal{C}(\tau \circ \sigma^{-1})$. Hence, $\sharp\mathcal{C}(\tau \circ \sigma^{-1}) \leq \sharp\mathcal{C}(l_{\text{pos}}(\tau) \circ \sigma^{-1})$. □

Lemma 5. *The Hamming distance H satisfies Requirement 1. In particular, $H(l_{\text{pos}}(\tau), \sigma) \leq H(\tau, \sigma)$ for every local permutation $\sigma \in \text{Ext}(\kappa)$ and permutation $\tau \in \text{Perm}(\mathcal{D})$.*

Proof. Let $x \notin \mathcal{H}(\tau, \sigma)$ be a candidate where τ and σ agree, i. e., $\sigma(x) = \tau(x)$. Since σ is local, x is not moved by l_{pos} as x is placed at a position belonging to its bucket, i. e., $x \notin \mathcal{A}_\tau$. Hence, $x \notin \mathcal{H}(l_{\text{pos}}(\tau), \sigma)$ and thus, $\mathcal{H}(l_{\text{pos}}(\tau), \sigma) \subseteq \mathcal{H}(\tau, \sigma)$. □

Lemma 6. *For every $p \in \mathbb{N} \setminus \{0\}$ the raised Minkowski distance $\hat{F}_p$ satisfies Requirement 1. In particular, $\hat{F}_p(l_{\text{pos}}(\tau), \sigma) \leq \hat{F}_p(\tau, \sigma)$ for every local permutation $\sigma \in \operatorname{Ext}(\kappa)$ and permutation $\tau \in \operatorname{Perm}(\mathcal{D})$.*

Proof. Let $x \in \mathcal{D}$. If $\tau(x) = \sigma(x)$ then $l_{\text{pos}}(\tau)(x) = \sigma(x)$, since $x \notin \mathcal{A}_\tau$. Otherwise, $|\tau(x) - \sigma(x)| \geq 1$ implies $|\tau(x) - \sigma(x)|^p \geq 1$, but $|l_{\text{pos}}(\tau)(x) - \sigma(x)| \leq 1$. In both cases $|l_{\text{pos}}(\tau)(x) - \sigma(x)|^p \leq |\tau(x) - \sigma(x)|^p$. □

Lemma 7. *MR under the raised Minkowski distance $\hat{F}_p$ for $p \in \mathbb{N} \setminus \{0\}$ and under the Hamming distance H satisfies Requirement 2.*

Proof. Let $\sigma, \tau \in \operatorname{Ext}(\kappa)$ be local permutations. Recall that $\mathcal{K}(\sigma, \tau) \subseteq \{\, \mathcal{B}_i \colon i \in \{1, \ldots, n\} \,\}$ as σ and τ agree on the order of candidates in different buckets. Hence, $|\tau(x) - \sigma(x)| = |\tau(y) - \sigma(y)| = 1$ for every bucket $\{x, y\} \in \mathcal{K}(\sigma, \tau)$, i. e., both x and y contribute 1 to the distance. Members of the remaining buckets $\{x, y\} \in \{\, \mathcal{B}_i \colon i \in \{1, \ldots, n\} \,\} \setminus \mathcal{K}(\sigma, \tau)$ contribute 0 to the distance. Thus, $d(\sigma, \tau) = |\mathcal{K}(\sigma, \tau)| = K(\sigma, \tau)$. □

Lemma 8. *The Ulam distance U satisfies Requirement 1. In particular, $U(l_{\text{ord}}(\tau), \sigma) \leq U(\tau, \sigma)$ for every local permutation $\sigma \in \operatorname{Ext}(\kappa)$ and permutation $\tau \in \operatorname{Perm}(\mathcal{D})$.*

Proof. Let $(x_1, \ldots, x_l)$ be a *longest* common subsequence of τ and σ with $l = \operatorname{lcs}(\sigma, \tau)$. As $\sigma \in \operatorname{Ext}(\kappa)$, all elements $x_1, \ldots, x_l$ are ordered by both σ and τ according to κ. Hence, $(x_1, \ldots, x_l)$ is also a common subsequence of $l_{\text{ord}}(\tau)$ and σ and thus, $\operatorname{lcs}(l_{\text{ord}}(\tau), \sigma) \geq \operatorname{lcs}(\tau, \sigma)$. Consequently, $U(l_{\text{ord}}(\tau), \sigma) = |\mathcal{D}| - \operatorname{lcs}(l_{\text{ord}}(\tau), \sigma) \leq |\mathcal{D}| - \operatorname{lcs}(\tau, \sigma) = U(\tau, \sigma)$. □

Lemma 9. *The Ulam distance U satisfies Requirement 2.*

Proof. Let $\sigma, \tau \in \operatorname{Ext}(\kappa)$ be local permutations and let $a_i, b_i \in \mathcal{B}_i$ for some bucket $\mathcal{B}_i$. Then $\{a_i, b_i\}$ might be a dirty pair, but a_i and b_i are not part of any other dirty pair, since σ and τ agree on the order of candidates in different buckets. Hence, at least one of a_i or b_i appears in every longest common subsequence of σ and τ. Both a_i and b_i occur in every longest common subsequence if and only if $\{a_i, b_i\} \notin \mathcal{K}(\sigma, \tau)$. Hence, for $n = |\mathcal{D}|$ candidates and $\frac{n}{2}$ buckets, we obtain $n - \operatorname{lcs}(\sigma, \tau) = n - (n - |\mathcal{K}(\sigma, \tau)|) = K(\sigma, \tau)$. □

For the proof of the following lemma, we extend the notion of the set of dirty pairs from permutations to strings which may contain duplicates. For strings s and t over an alphabet $\mathcal{D}$ of size $2n$, define the set of dirty character pairs $\{x, y\}$ such that in s some occurrence of x is before some occurrence of y and conversely in t:

$$\mathcal{K}(s,t) = \{\{x,y\} \subseteq \mathcal{D} : \exists i, j, i', j' \in \{1, \ldots, 2n\} : \\ s(i) = t(i') = x \wedge s(j) = t(j') = y \wedge i < j \wedge i' > j'\}\,.$$

Observe that this generalization is backwards-compatible: If the strings s and t are duplicate-free, i. e., $\sigma = s^{-1}$ and $\tau \in t^{-1}$ are permutations on $\mathcal{D}$, then $\mathcal{K}(s,t) = \mathcal{K}(\sigma, \tau)$. The extension to strings is pessimistic in the sense that two characters x and y attempt to belong to $K(s,t)$ if they occur in different orders. As a consequence, a dirty pair cannot be resolved by inserting more occurrences of x or y into the strings.

Lemma 10. *The Damerau-Levenshtein and Swap-and-Mismatch distances satisfy Requirement 1. In particular, $d(l_{\mathrm{ord}}(\tau), \sigma) \leq d(\tau, \sigma)$ for every local permutation $\sigma \in \mathrm{Ext}(\kappa)$, permutation $\tau \in \mathrm{Perm}(\mathcal{D})$, and $d \in \{D, S\}$.*

Proof. For a string t define the set $\mathcal{X}(t) = \{\mathcal{B}_i : i \in \{1, \ldots, n\}\} \cap \mathcal{K}(\sigma^{-1}, t)$ of buckets forming a dirty pair. Hence, every character x occurs in at most one dirty pair $\mathcal{B}_i \in \mathcal{X}(t)$ for some $i \in \{1, \ldots, n\}$.

Consider a shortest sequence $(s_1 = \tau^{-1}, s_2, \ldots, s_{k+1} = \sigma^{-1})$ transforming τ into σ with k operations allowed for d. We examine the size of $\mathcal{X}(s_i)$ during the transformation process. In the end, $\mathcal{X}(s_{k+1}) = \emptyset$, since s_{k+1} and σ^{-1} are identical and duplicate-free.

However, for every applied operation, $\mathcal{X}(s_i)$ may decrease by at most one dirty pair: Inserting a character cannot decrease the number of dirty pairs, while substituting, deleting, or swapping adjacent characters eliminates at most one dirty pair as the buckets are disjoint. Thus, $|\mathcal{X}(\tau^{-1})|$ is a lower bound on the number of needed operations.

Now consider the local permutation $l_{\mathrm{ord}}(\tau)$. We have $\mathcal{X}(l_{\mathrm{ord}}(\tau)^{-1}) = \mathcal{X}(\tau^{-1})$, since l_{ord} preserves the order of elements from the same bucket. Again, $|\mathcal{X}(l_{\mathrm{ord}}(\tau)^{-1})|$ is a lower bound on the number of needed operations to transform $l_{\mathrm{ord}}(\tau)$ into σ. However, we actually need only $|\mathcal{X}(l_{\mathrm{ord}}(\tau)^{-1})|$ operations, since in $l_{\mathrm{ord}}(\tau)$ candidates from each dirty pair are placed next to each other, thus can be resolved by a single swap. Hence, $d(l_{\mathrm{ord}}(\tau), \sigma) = |\mathcal{X}(l_{\mathrm{ord}}(\tau)^{-1})| = |\mathcal{X}(\tau^{-1})| \leq d(\tau, \sigma)$. □

Lemma 11. *The Damerau-Levenshtein and Swap-and-Mismatch distances satisfy Requirement 2.*

Proof. Let $\sigma, \tau \in \text{Ext}(\kappa)$ be local permutations and $d \in \{D, S\}$. As shown in the proof of Lemma 10, iteratively resolving dirty pairs by a swap of adjacent characters is a shortest sequence of transforming τ into σ. Thus, $d(\tau, \sigma) = K(\tau, \sigma)$. $\square$

Lemma 12. *The Lee distance L satisfies Requirement 1, i. e., $L(l_{\text{pos}}(\tau), \sigma) \leq L(\tau, \sigma)$ for every local permutation $\sigma \in \text{Ext}(\kappa)$ and permutation $\tau \in \text{Perm}(\mathcal{D})$.*

Proof. Let $\sigma \in \text{Ext}(\kappa)$ be a local permutation, $\tau \in \text{Perm}(\mathcal{D})$ a permutation on a candidate set $\mathcal{D}$ of size n, and $x \in \mathcal{D}$. If $\tau(x) = \sigma(x)$, then $l_{\text{pos}}(\tau)(x) = \sigma(x)$, since $x \notin \mathcal{A}_\tau$. Otherwise, both $|\tau(x) - \sigma(x)| \geq 1$ and $n - |\tau(x) - \sigma(x)| \geq 1$. In both cases $|l_{\text{pos}}(\tau)(x) - \sigma(x)| \leq |\tau(x) - \sigma(x)|$ and $|l_{\text{pos}}(\tau)(x) - \sigma(x)| \leq n - |\tau(x) - \sigma(x)|$. Thus, $L(l_{\text{pos}}(\tau), \sigma) \leq L(\tau, \sigma)$. $\square$

Lemma 13. *The Lee distance L satisfies Requirement 2.*

Proof. Let $\sigma, \tau \in \text{Ext}(\kappa)$ be local permutations and $n = |\mathcal{D}| > 1$. Then $|\tau(x) - \sigma(x)| \leq 1 \leq n - |\tau(x) - \sigma(x)|$ for every candidate $x \in \mathcal{D}$. Hence, $L(\sigma, \tau) = \sum_{x \in \mathcal{D}} \min\{|\tau(x) - \sigma(x)|, n - |\tau(x) - \sigma(x)|\} = \sum_{x \in \mathcal{D}} |\tau(x) - \sigma(x)| = F_1(\sigma, \tau) = 2\,|\mathcal{K}(\sigma, \tau)|$ by Lemma 7. $\square$

Theorem 3. *Requirements 1 and 2 are satisfied by the Kendall tau, Cayley, Hamming, Ulam, Damerau-Levenshtein, Swap-and-Mismatch, Lee, and Minkowski distances F_p for $p \in \mathbb{N} \setminus \{0\}$. Thus, MR is* **NP***-complete under these distances.*

A notable consequence is the dichotomy between the sum and the maximum versions of the rank aggregation problem, in particular for the Spearman footrule distance.

Corollary 2. *For the Minkowski distances F_p and $p \in \mathbb{N} \setminus \{0\}$ the common rank aggregation problem is efficiently solvable, whereas the maximum rank aggregation problem MR is* **NP***-complete.*

Proof. The common rank aggregation problem can be solved by weighted bipartite matching, where the weights $w_{x,i}$ express the cost of placing x at position i [7], and **NP**-completeness of MR follows from Theorem 3. $\square$

Since the Minimum distance does not satisfy Requirement 2, we provide a different reduction from Hitting String.

Definition 4 (Hitting String [34])**.**
Instance: $n \in \mathbb{N}$*, a list* $s_1, \ldots, s_m \in \{0, 1, *\}^n$ *of* m *strings of length* n*.*
Question: Does there exist a string $t \in \{0, 1\}^n$ *such that each string* s_j is hit *by* t *in at least one position, i. e.,* $\forall j \in \{1, \ldots, m\} : \exists i \in \{1, \ldots, n\} : s_j(i) = t(i)$*.*

Theorem 4. *MR under the Minimum distance* $F_{-\infty}$ *is* **NP***-complete even for zero distance* $k = 0$*.*

Proof. There is a consensus permutation $\tau \in \mathrm{Perm}(\mathcal{D})$ with $\max_{j=1}^{m} \min_{x \in \mathcal{D}} |\sigma_j(x) - \tau(x)| = 0$ if and only if for every σ_j there is a candidate x such that $\sigma_j(x) = \tau(x)$. Then τ *hits* σ_j *at position* $\tau(x)$ and τ is a *hitting consensus*.

First, we show how to construct an instance with $2n$ voters of length n which has no hitting consensus. Let $\mathcal{D} = \{u_1, \ldots, u_n\}$ and $\sigma_1 : \mathcal{D} \to \{1, \ldots, n\} : u_i \mapsto i$. We obtain n *primary voters* $\sigma_1, \ldots, \sigma_n$ by rotating σ_1, i. e., for every $j \in \{1, \ldots, n\}$ let $\sigma_j(u_i) = (i+j-2) \mod n+1$. Additionally, we introduce *secondary voters* $\sigma'_1, \ldots, \sigma'_n$ defined by $\sigma'_j = T_{1,2} \circ \sigma_j$. For instance if $\mathcal{D} = \{a, b, c, d, e\}$, then the list of voters is

$$\begin{array}{ll}
\sigma_1 = [abcde] & \sigma'_1 = [bacde] \\
\sigma_2 = [eabcd] & \sigma'_2 = [aebcd] \\
\sigma_3 = [deabc] & \sigma'_3 = [edabc] \\
\sigma_4 = [cdeab] & \sigma'_4 = [dceab] \\
\sigma_5 = [bcdea] & \sigma'_5 = [cbdea].
\end{array}$$

Assume for contradiction that this list of voters has a hitting consensus τ. Since there are n primary voters and no two primary voters place any candidate at the same position, every primary voter is hit at exactly one position and τ hits exactly one primary voter at position 1. Let σ be the primary voter hit at position 1 by a candidate x. Then τ cannot hit the secondary voter $\sigma' = T_{1,2} \circ \sigma$ at the positions 1 or 2 as $\tau(x) = \sigma(x) = 1 \neq 2 = \sigma'(x)$. Thus, it cannot hit σ' at all, since σ and σ' agree in all other positions $\{1, \ldots, n\} \setminus \{1, 2\}$, a contradiction. We call the above list of voters the n-*anti-pattern*. With this in mind, we reduce from the **NP**-complete Hitting String to MR under the Minimum distance.

As in the proof of Theorem 2 (see also [13]), let $\mathcal{D} = \bigcup_{i=1}^{n} \{a_i, b_i\}$ be the set of candidates and let $f : \{0, 1\}^n \to \mathrm{Perm}(\mathcal{D})$ with $f(s)(a_i) = 2i-1+s(i)$ and $f(s)(b_i) = 2i - s(i)$. For each string s_j, $j \in \{1, \ldots, m\}$, we introduce a list of voters Σ_j in two steps. The instance of MR is then the concatenation of all Σ_j

and $k = 0$. In the first step, we create a *template* $\rho_j : \mathcal{D} \to \{1, \dots, n\} \cup \{*\}$ from which the actual list is obtained in the second step. Let $\rho_j(a_i) = f(s_j)(a_i)$ and $\rho(b_i) = f(s_j)(b_i)$ if $s(i) \in \{0, 1\}$ and $\rho_j(a_i) = \rho_j(b_j) = *$, otherwise. If none of the strings s_j did contain $*$, then we could establish a one-to-one correspondence between a hitting consensus for voters $\rho_1, \dots, \rho_m$ and a hitting string for $s_1, \dots, s_m$ as in Theorem 2 and would be done. Let $\mathcal{U}_j = \{x \in \mathcal{D} : \rho_j = *\}$ be the set of candidates which are not assigned a position by ρ_j. In Hitting String the $*$ marks a position where an input string cannot be hit however the hitting string looks alike. We reproduce this situation for MR by making $2\,|\mathcal{U}_j|$ copies $\sigma_j^{(1)}, \dots, \sigma_j^{(2|\mathcal{U}_j|)}$ of ρ_j such that all copies agree on the candidates $\mathcal{D} \setminus \mathcal{U}_j$ but form a $|\mathcal{U}_j|$-anti-pattern if the candidate set is restricted to $\mathcal{U}_j$.

Suppose that t^* is a hitting string for $s_1, \dots, s_m$. Then $f(t^*)$ is a hitting consensus, since for every $j \in \{1, \dots, m\}$ there is an $i \in \{1, \dots, n\}$ with $t^*(i) = s_j(i)$, thus $f(t^*)(a_i) = \sigma_j(a_i)$. Conversely, suppose that τ^* is a hitting consensus. Consider the string $t^* \in \{0, 1\}^n$ defined by $t^*(i) = 0$ if $\tau^*(a_i) = 2i - 1 \vee \tau^*(b_i) = 2i$ and $t^*(i) = 1$, otherwise. For every $j \in \{1, \dots, m\}$ there must be a candidate $x \notin \mathcal{U}_j$ with $\tau(x) = \sigma_j^{(r)}(x)$ for all $\sigma_j^{(r)} \in \Sigma_j$, since they form a $|\mathcal{U}_j|$-anti-pattern when restricted to $\mathcal{U}_j$. The position of $x \in \mathcal{D} \setminus \mathcal{U}_j$ in all $\sigma_j^{(r)} \in \Sigma_j$ is identical and determined by s_j. Therefore, $x = a_i$ or $x = b_i$ for a position i where $s_j(i) \neq *$ and thus, $t^*(i) = s_j(i)$. Hence, t^* is a hitting string. □

5. Approximability

Due to the general **NP**-hardness of MR, one may ask for an approximation. In fact, there is a straightforward 2-approximation.

Lemma 14. *The associated minimization problem of MR is 2-approximable for any pseudometric d.*

Proof. Let $\tau^* \in \mathrm{Perm}(\mathcal{D})$ be the optimal consensus for the MR problem under pseudometric d with voters $\sigma_1, \dots, \sigma_m \in \mathcal{D}$. Then the *pick-a-perm* method [1] with $\tau = \sigma_j$ for any $j \in \{1, \dots, m\}$ yields a 2-approximation, since for all $i \in \{1, \dots, m\}$ we have

$$
\begin{aligned}
d(\sigma_i, \tau) &\leq d(\sigma_i, \tau^*) + d(\tau^*, \tau) \leq 2 \cdot \max\{d(\sigma_i, \tau^*), d(\tau^*, \tau)\} \\
&\leq 2 \cdot \max_{j=1}^{m} d(\sigma_j, \tau^*)\,.
\end{aligned}
$$

□

Note that this approximation ratio for pick-a-perm is tight for all metrics satisfying Requirements 1 and 2. For instance, consider the voters $f(\texttt{"1000...")}$, $f(\texttt{"0100...")}$, $f(\texttt{"0010...")}$ with f as defined in the proof of Theorem 2. The distance between each pair of voters is $2c$, while the optimal consensus is $f(\texttt{"0000...")}$ with distance c.

6. Fixed-Parameter Tractability

The reduction in Sect. 4 demonstrates a close relationship between Closest Binary String and MR. We strengthen this observation by extending a fixed parameter algorithm for Closest Binary String [35, 36] such that it can be applied to MR under several metrics. A similar approach has been developed by Schwarz [14]. Again we pursue a general schema which captures several distances. For an introduction to fixed-parameter tractability see [36, 37].

The notion of the *modification set* $M(\tau, \sigma) \subseteq \mathrm{Perm}(\mathcal{D})$ is at the heart of our schema. Intuitively, it captures the idea of going "one step" from τ to σ, i. e., the set consists of permutations near τ which are slightly closer to σ. The structure of the modification set must be chosen separately for each metric d. We state a sufficient condition, which we call the δ*-improving* of M, such that the algorithm actually finds a k-consensus.

Requirement 3 (δ-improving)**.** *Let $\delta \in \mathbb{N} \setminus \{0\}$, $\sigma, \tau, \tau^* \in \mathrm{Perm}(\mathcal{D})$, and $k \in \mathbb{N}$ such that $d(\tau^*, \sigma) \leq k$ and $d(\tau, \tau^*) \leq k$. If $k < d(\tau, \sigma) \leq 2k$, then there exists a permutation $\tau' \in M(\tau, \sigma)$ in the modification set such that $d(\tau', \tau^*) \leq d(\tau, \tau^*) - \delta$.*

In other words, by approaching distant voters σ from τ, Requirement 3 guarantees that at least one permutation in $M(\tau, \sigma)$ is closer to the (unknown) k-consensus τ^* if such a k-consensus exists. Hence, in Algorithm 1 we start with $\tau = \sigma_1$ and recursively test all permutations of the modification set until τ actually reaches τ^* or no τ^* exists within a search depth of k.

Lemma 15. *Let $\sigma_1, \ldots, \sigma_m \in \mathrm{Perm}(\mathcal{D})$ be a list of m voters and $k \in \mathbb{N}$ be a non-negative integer. Suppose that there is a k-consensus τ^*, i. e., $\max_{j=1}^{m} d(\tau^*, \sigma_j) \leq k$. If M is δ-improving, then at recursion depth i,* search *in Algorithm 1 has either already found a k-consensus, or is called at least once with a parameter τ such that $d(\tau, \tau^*) \leq k - \delta i$.*

```
   Input: Voters σ_1, ..., σ_m ∈ Perm(𝒟), bound k ∈ ℕ.
   Output: k-consensus τ* ∈ Perm(𝒟) or reject.
1  search(σ_1, k);
2  function search(τ, Δk)
3      if ∀j ∈ {1, ..., m} : d(τ, σ_j) ≤ k then return τ;
4      if ∃j ∈ {1, ..., m} : d(τ, σ_j) > k + Δk then return ⊥;
5      if Δk > 0 then
6          let j ∈ {1, ..., m} such that k < d(τ, σ_j) ≤ k + Δk;
7          foreach τ' ∈ M(τ, σ_j) do
8              ρ ← search(τ', Δk − δ);
9              if ρ ≠ ⊥ then return ρ;
10     return ⊥;
```

Algorithm 1: Fixed-parameter algorithm for MR

Proof. The proof is by induction on the recursion depth i. The induction basis $d(\tau, \tau^*) \leq k - 0$ holds, since $\tau = \sigma_1$ in depth 0. For the induction step, suppose the program has not yet found the solution and that at recursion depth $i \leq \lceil \frac{k}{\delta} \rceil$ search is called with τ' having $d(\tau, \tau^*) \leq k - \delta i$. If $d(\tau, \sigma_j) \leq k$ for all $j \in \{1, \ldots, m\}$, we have found a k-consensus and are done. Otherwise, there is a j such that $d(\tau, \sigma_j) > k$. The break condition in line 4 does not hold, since $d(\tau, \sigma_j) \leq d(\tau, \tau^*) + d(\tau^*, \sigma_j) \leq k - \delta i + k = \Delta k + k$. As τ' iterates over $M(\tau, \sigma_j)$, by Requirement 3 there is at least one iteration where search is called with a τ' where $d(\tau', \tau^*) \leq k - \delta i - \delta$. $\square$

Theorem 5. *If M is δ-improving, then Algorithm 1 finds a k-consensus τ^* or correctly reports that no such consensus exists. Its running time is $\mathcal{O}((f(k))^{\lceil \frac{k}{\delta} \rceil} \cdot g(k, n))$, where $f(k)$ is the maximum size of the constructed modification sets and $g(k, n)$ is the time required for the construction of a modification set.*

Proof. The recursion depth is bounded by $\lceil \frac{k}{\delta} \rceil$ and the branching factor is limited by the maximum size of the modification set. The running time is worst if no k-consensus exists, in which case search returns the empty set. Otherwise, suppose that τ^* is a k-consensus. Then, by Lemma 15, search finds a different k-consensus or is eventually called with a τ such that $d(\tau, \tau^*) = 0$ which implies $\tau = \tau^*$. $\square$

For fixed-parameter results it remains to construct a suitable modification set for each distance.

Lemma 16. *The modification set* $M(\tau, \sigma) = \{T_{\tau(x),\tau(y)} \circ \tau : \{x, y\} \in \mathcal{K}(\tau, \sigma)\}$ *is 1-improving under the Kendall tau distance* K*.*

Proof. Let $k \in \mathbb{N}$ and $\sigma, \tau, \tau^* \in \mathrm{Perm}(\mathcal{D})$ such that $K(\tau^*, \sigma) \leq k < K(\tau, \sigma) \leq 2k$. Note that by Lemma 1, $d(T_{\tau(x),\tau(y)} \circ \tau, \tau^*) < d(\tau, \tau^*)$ for each dirty pair $\{x, y\} \in \mathcal{K}(\tau, \sigma) \cap \mathcal{K}(\tau, \tau^*)$. Thus, it suffices to show that $\mathcal{K}(\tau, \sigma) \cap \mathcal{K}(\tau, \tau^*) \neq \emptyset$.

Assume for contradiction that $\mathcal{K}(\tau, \sigma)$ and $\mathcal{K}(\tau, \tau^*)$ are disjoint. Let $\{x, y\} \in \mathcal{K}(\tau, \sigma)$. As $\{x, y\} \notin \mathcal{K}(\tau, \tau^*)$, we know that τ and τ^* agree on the relative order of x and y, which implies that $\{x, y\} \in \mathcal{K}(\sigma, \tau^*)$. Hence, $\mathcal{K}(\tau, \sigma) \subseteq \mathcal{K}(\sigma, \tau^*)$. Now let $\{x, y\} \in \mathcal{K}(\tau, \tau^*)$. As $\{x, y\} \notin \mathcal{K}(\tau, \sigma)$, τ and σ agree on the relative order of x and y, implying $\{x, y\} \in \mathcal{K}(\sigma, \tau^*)$. Hence, $\mathcal{K}(\tau, \tau^*) \subseteq \mathcal{K}(\sigma, \tau^*)$. We conclude that $K(\sigma, \tau^*) = |\mathcal{K}(\sigma, \tau^*)| \geq |\mathcal{K}(\tau, \sigma)| + |\mathcal{K}(\tau, \tau^*)| \geq k + 1$, a contradiction. $\square$

Corollary 3. *MR under the Kendall tau distance* K *can be computed in* $\mathcal{O}((2k)^k \cdot (mn \log n + k)\})$ *time.*

Proof. Consider the modification set of Lemma 16, whose size is $|M(\tau, \sigma)| = |\mathcal{K}(\tau, \sigma)| = K(\tau, \sigma) \leq 2k$. The distance of two permutations can be computed in $\mathcal{O}(n \log n)$ time [31]. Hence, lines 3, 4 and 6 of Algorithm 1 need $\mathcal{O}(mn \log n)$ time. For efficiency reasons, we represent the modification set $M(\tau, \sigma)$ only implicitly by the set $\mathcal{K}(\tau, \sigma)$ of at most $2k$ dirty pairs, which can be computed in $\mathcal{O}(n \log n + k)$ time [11]. We iterate τ' over $M(\tau, \sigma)$ by transposing the next dirty pair in $\mathcal{K}(\tau, \sigma)$, descent recursively, and undo the transposition after the recursive call returns. Thus, excluding the recursion, the loop requires $\mathcal{O}(n \log n + k)$ time. $\square$

Lemma 17. *Let* σ *and* $\tau \in \mathrm{Perm}(\mathcal{D})$ *be two permutations which disagree on the position of a displaced candidate* x*, i. e.,* $x \in \mathcal{H}(\sigma, \tau)$*. Then* $H(\sigma, T_{\sigma(x),\tau(x)} \circ \tau) < H(\sigma, \tau)$*.*

Proof. Let $y \in \mathcal{D}$ such that $\tau(y) = \sigma(x)$. Note that $y \in \mathcal{H}(\sigma, \tau)$ and the transposition of x and y in τ does not affect other candidates. Thus, $\mathcal{H}(\sigma, T_{\sigma(x),\tau(x)} \circ \tau) = \mathcal{H}(\sigma, \tau) \setminus \{x\}$ or even $\mathcal{H}(\sigma, T_{\sigma(x),\tau(x)} \circ \tau) = \mathcal{H}(\sigma, \tau) \setminus \{x, y\}$ if $\tau(x) = \sigma(y)$. $\square$

Lemma 18. *The modification set* $M(\tau,\sigma) = \{T_{\tau(x),\sigma(x)} \circ \tau : x \in \mathcal{H}(\tau,\sigma)\}$ *is 1-improving under the Hamming distance* H*.*

Proof. Let $k \in \mathbb{N}$ and $\sigma, \tau, \tau^* \in \mathrm{Perm}(\mathcal{D})$ such that $H(\tau^*, \sigma) \leq k < H(\tau,\sigma) \leq 2k$. The size of the modification set is $|M(\tau,\sigma)| = |\mathcal{H}(\tau,\sigma)| = H(\tau,\sigma) \leq 2k$. σ and τ^* agree in the position of at least $|D| - k$ candidates. As $H(\tau,\sigma) > k$, there is at least one candidate x with $\tau(x) \neq \sigma(x) = \tau^*(x)$. Hence, $H(T_{\tau(x),\sigma(x)} \circ \tau, \tau^*) < H(\tau,\tau^*)$ by Lemma 17. □

Corollary 4. *MR under the Hamming distance* H *can be computed in* $\mathcal{O}((2k)^k \cdot mn)$ *time.*

Proof. Consider the modification set of Lemma 18. The Hamming distance between two permutations can be computed in linear time. Thus, lines 3, 4 and 6 of Algorithm 1 need $\mathcal{O}(mn)$ time. Similarly to the proof of Corollary 3, the iteration of τ' over the modification set $M(\tau,\sigma)$ with $|M(\tau,\sigma)| \leq 2k$ is done in place and needs only $\mathcal{O}(k)$ time. □

Lemma 19. *The modification set* $M(\tau,\sigma) = \{T_{\tau(x),i} \circ \tau : x \in \mathcal{H}(\tau,\sigma) \wedge i \in \{\tau(x) + j \cdot \mathrm{sgn}(\sigma(x) - \tau(x)) : j \in \{1, \ldots, \lceil k^{\frac{1}{p}} \rceil\}\} \cap \{1, \ldots, n\}\}$ *is* $(p+1)$*-improving under the raised Minkowski distance* $\hat{F}_p$ *for* $p \in \mathbb{N} \setminus \{0\}$*.*

Proof. Let $k \in \mathbb{N}$ and $\sigma, \tau, \tau^* \in \mathrm{Perm}(\mathcal{D})$ such that $\hat{F}_p(\tau^*, \sigma) \leq k < \hat{F}_p(\tau,\sigma) \leq 2k$. We take every displaced candidate $x \in \mathcal{H}(\tau,\sigma)$ and try all possibilities to transpose it with candidates placed at most k positions to its right or left, depending on whether $\sigma(x) > \tau(x)$ or $\sigma(x) < \tau(x)$, respectively. Suppose we have a candidate $x \in \mathcal{H}(\tau,\sigma)$ with $|\sigma(x) - \tau^*(x)| < |\sigma(x) - \tau(x)|$. There must be at least one such candidate, since $\hat{F}_p(\tau^*, \sigma) < \hat{F}_p(\tau,\sigma)$. W. l. o. g. assume $\sigma(x) > \tau(x)$. Otherwise, the following arguments apply symmetrically. Let $\mathcal{Y} = \{\tau^{*-1}(i) : i \leq \tau(x)\}$ be the set of candidates which are placed in τ^* to the left of or on the same position where x is placed in τ. As $\tau^*(x) > \tau(x)$, $x \notin \mathcal{Y}$, so by a counting argument there must be some $y \in \mathcal{Y}$ with $\tau(y) > \tau(x)$. We know that $\tau' = T_{\tau(x),\tau(y)} \circ \tau$ is contained in the modification set because $\tau(y) - \tau^*(y) \leq k^{\frac{1}{p}}$ due to $\hat{F}_p(\tau,\tau^*) \leq k$ by Requirement 3. We distinguish two cases whether or not $\tau(y) \leq \tau^*(x)$.

Case 1: $\tau^*(y) \leq \tau(x) < \tau(y) \leq \tau^*(x)$.. Then both $\tau^*(x) - \tau'(x) = \tau^*(x) - \tau(y) < \tau^*(x) - \tau(x)$ and $\tau'(y) - \tau^*(y) = \tau(x) - \tau^*(y) < \tau(y) - \tau^*(y)$. Hence, by the Binomial Theorem, $|\tau(x) - \tau^*(x)|^p - |\tau'(x) - \tau^*(x)|^p =$

$$\underbrace{(|\tau(x) - \tau^*(x)| - |\tau'(x) - \tau^*(x)|)}_{\geq 1} \cdot \sum_{i=0}^{p-1} \underbrace{|\tau(x) - \tau^*(x)|^i}_{\geq 1} \cdot \underbrace{|\tau'(x) - \tau^*(x)|^{p-i-1}}_{\geq 1}$$

and thus, $|\tau'(x) - \tau^*(x)|^p \leq |\tau(x) - \tau^*(x)|^p - p$. We obtain $|\tau'(y) - \tau^*(y)|^p \leq |\tau(y) - \tau^*(y)|^p - p$ symmetrically. In sum $|\tau'(x) - \tau^*(x)|^p + |\tau'(y) - \tau^*(y)|^p \leq |\tau(x) - \tau^*(x)|^p + |\tau(y) - \tau^*(y)|^p - 2p$.

Case 2: $\tau^*(y) \leq \tau(x) < \tau^*(x) < \tau(y)$.. Then $\tau'(x) - \tau^*(x) + \tau'(y) - \tau^*(y) = \tau(y) - \tau^*(x) + \tau(x) - \tau^*(y) < \tau(y) - \tau^*(y)$. By the Binomial Theorem we derive

$$(\tau'(x) - \tau^*(x) + \tau'(y) - \tau^*(y))^p \leq (\tau(y) - \tau^*(y))^p - p$$
$$|\tau'(x) - \tau^*(x)|^p + |\tau'(y) - \tau^*(y)|^p \leq |\tau(y) - \tau^*(y)|^p - p + \underbrace{|\tau(x) - \tau^*(x)|^p}_{\geq 1} - 1$$

Recall that the positions of candidates $\mathcal{D} \setminus \{x, y\}$ are unaffected. Hence, in both cases $\hat{F}_p(\tau', \tau^*) \leq \hat{F}_p(\tau, \tau^*) - (p + 1)$. $\square$

Corollary 5. *MR under the Minkowski distance F_p for $p \in \mathbb{N} \setminus \{0\}$ can be computed in $\mathcal{O}((2k^{p+1})^{\lceil \frac{k^p}{p+1} \rceil} \cdot mn)$ time.*

Proof. Let $\hat{k} = k^p$. Finding a k-consensus for F_p is equivalent to finding a $\hat{k}$-consensus for $\hat{F}_p$. Consider the modification set of Lemma 19. Its size is $|M(\tau, \sigma)| \leq 2\hat{k}^{1+\frac{1}{p}}$, since there are most $2\hat{k}$ displaced candidates which are each tested on at most $\hat{k}^{\frac{1}{p}}$ positions. The Minkowski distance between two permutations can be computed in linear time. Thus, lines 3, 4 and 6 of Algorithm 1 need $\mathcal{O}(mn)$ time. Finding the up to $2\hat{k}$ displaced candidates to build the modification set needs $\mathcal{O}(n)$ time. Each displaced candidate is tested on $\hat{k}^{\frac{1}{p}}$ positions. Then the total running time is in $\mathcal{O}((2\hat{k}^{1+\frac{1}{p}})^{\lceil \frac{\hat{k}}{p+1} \rceil} \cdot mn) = \mathcal{O}((2k^{p+1})^{\lceil \frac{k^p}{p+1} \rceil} \cdot mn)$. $\square$

There are tractable algorithms for Closest Binary String parameterizing the number of strings m [36]. However, parameterizing MR by the number of voters m does not lead to efficient algorithms, since MR under the Kendall tau distance is **NP**-hard for $m = 4$ [7, 11].

Note that the **NP**-hardness of MR under the Minimum distance even for $k = 0$ implies that this problem is not fixed-parameter tractable by k unless $\mathbf{P} = \mathbf{NP}$.

7. Conclusion

We explored the complexity of MR using a general schema which enables us to establish sufficient requirements for metrics under which MR is **NP**-complete and fixed-parameter tractable. Considering **NP**-hardness, the Requirements 1 and 2 may also hold for other distances. Considering fixed parameter tractability suitable modification sets (Requirement 3) must be found. An open field are better approximation ratios and the extension of MR for partial orders [4, 5, 19].

References

[1] N. Ailon, Aggregation of partial rankings, p-ratings and top-m lists, Algorithmica 57 (2) (2010) 284–300.

[2] N. Ailon, M. Charikar, A. Newman, Aggregating inconsistent information: Ranking and clustering, J. ACM 55 (5) (2008) 23:1–23:27.

[3] N. Betzler, M. R. Fellows, J. Guo, R. Niedermeier, F. A. Rosamond, Fixed-parameter algorithms for Kemeny rankings, Theor. Comput. Sci. 410 (8) (2009) 4554–4570.

[4] F. J. Brandenburg, A. Gleißner, A. Hofmeier, Comparing and aggregating partial orders with Kendall tau distances, Discrete Math. Algorithms Appl. 5 (2) (2013) .

[5] F. J. Brandenburg, A. Gleißner, A. Hofmeier, The nearest neighbor Spearman footrule distance for bucket, interval, and partial orders, J. Comb. Optim. 26 (2) (2013) 310–332.

[6] J. J. Bartholdi, C. A. Tovey, M. A. Trick, Voting schemes for which it can be difficult to tell who won the election, Soc. Choice Welfare 6 (2) (1989) 157–165.

[7] C. Dwork, R. Kumar, M. Naor, D. Sivakumar, Rank aggregation methods for the web, in: Proc. World Wide Web Conference, WWW 2010, 2001, pp. 613–622.

[8] C. Kenyon-Mathieu, W. Schudy, How to rank with few errors, in: Proc. Symposium on Theory of Computing, STOC 2007, ACM, 2007, pp. 95–103.

[9] M. Karpinski, W. Schudy, Faster algorithms for feedback arc set tournament, Kemeny rank aggregation and betweenness tournament, in: Proc. International Symposium on Algorithms and Computation, ISAAC 2010, Vol. 6506 of LNCS, Springer, 2010, pp. 3–14.

[10] J. G. Kemeny, Mathematics without numbers, Daedalus 88 (1959) 577–591.

[11] T. Biedl, F. J. Brandenburg, X. Deng, On the complexity of crossings in permutations, Discrete Math. 309 (7) (2009) 1813–1823.

[12] P. Diaconis, R. L. Graham, Spearman's footrule as a measure of disarray, J. Roy. Statist. Soc. B 39 (2) (1977) 262–268.

[13] V. Y. Popov, Multiple genome rearrangement by swaps and by element duplications, Theor. Comput. Sci. 385 (2007) 115–126.

[14] N. Schwarz, Rank aggregation by criteria: Minimizing the maximum Kendall-τ distance, Master's thesis, Friedrich-Schiller-Universität Jena (2009).

[15] J. J. Sylvester, A question in the geometry of situation, Quart. J. Math. 1 (1857) 79.

[16] M. I. Shamos, D. Hoey, Closest-point problems, in: Proc. Symposium on Foundations of Computer Science, FOCS 1975, SFCS '75, IEEE, 1975, pp. 151–162.

[17] R. L. Francis, J. A. White, Facility Layout and Location: an Analytical Approach, International Industrial and Systems Engineering Series, Prentice-Hall, 1974.

[18] D. Gusfield, Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology, Cambridge University Press, 1997.

[19] G. Fertin, A. Labarre, I. Rusu, É. Tannier, S. Vialette, Combinatorics of Genome Rearrangements, MIT Press, 2009.

[20] R. Fagin, R. Kumar, D. Sivakumar, Comparing top k lists, SIAM J. Discrete Math. 17 (1) (2003) 134–160.

[21] A. Caprara, Sorting permutations by reversals and Eulerian cycle decompositions, SIAM J. Discrete Math. 12 (1999) 91–110.

[22] L. Bulteau, G. Fertin, I. Rusu, Sorting by transpositions is difficult, SIAM J. Discrete Math. 26 (3) (2012) 1148–1180.

[23] C. Bachmaier, F. J. Brandenburg, A. Gleißner, A. Hofmeier, On maximum rank aggregation problems, in: T. Lecroq, L. Mouchard (Eds.), Proc. International Workshop on Combinatorial Algorithms, IWOCA 13, Vol. 8288 of LNCS, Springer, 2013, pp. 14–27.

[24] R. Fagin, R. Kumar, M. Mahdian, D. Sivakumar, E. Vee, Comparing partial rankings, SIAM J. Discrete Math. 20 (3) (2006) 628–648.

[25] D. E. Critchlow, Metric Methods for Analyzing Partially Ranked Data, Vol. 34 of LNS, Springer, 1985.

[26] M. Deza, T. Huang, Metrics on permutations, a survey, J. Combin. Inform. System Sci. 23 (1998) 173–185.

[27] V. I. Levenshtein, Binary codes capable of deletions, insertions, and reversals, Sov. Phys. Dokl. 10 (8) (1966) 707–710.

[28] B. J. Oommen, R. K. S. Loke, Pattern recognition of strings with substitutions, insertions, deletions and generalized transpositions, Pattern Recogn. 30 (1997) 789–800.

[29] A. Amir, E. Eisenberg, E. Porat, Swap and mismatch edit distance, Algorithmica 45 (2006) 109–120.

[30] C. Y. Lee, Some properties of nonbinary error-correcting codes, IRE T. Inform. Theor. 4 (2) (1958) 77–82.

[31] W. R. Knight, A computer method for calculating Kendall's tau with ungrouped data, J. Amer. Statist. Assoc. 61 (314) (1966) 436–439.

[32] L. Bergroth, H. Hakonen, T. Raita, A survey of longest common subsequence algorithms, in: Proc. String Processing and Information Retrieval, SPIRE 2000, IEEE, 2000, pp. 39–48.

[33] M. Frances, A. Litman, On covering problems of codes, Theor. Comput. Syst. 30 (2) (1997) 113–119.

[34] R. Fagin, Generalized first-order spectra and polynomial-time recognizable sets, in: R. Karp (Ed.), Complexity of Computation, Vol. 7 of SIAM-AMS Proceedings, ACM, 1974, pp. 43–73.

[35] J. Gramm, R. Niedermeier, P. Rossmanith, Fixed-parameter algorithms for CLOSEST STRING and related problems, Algorithmica 37 (1) (2003) 25–42.

[36] R. Niedermeier, Invitation to Fixed-Parameter Algorithms, Oxford University Press, 2006.

[37] R. G. Downey, M. R. Fellows, Fundamentals of Parameterized Complexity, Texts in Computer Science, Springer, 2013.