Lower Bounds on Kemeny Rank Aggregation with Non-Strict Rankings

Sina Akbari
School of Computing and Augmented Intelligence
Arizona State University
Tempe, Arizona, United States
sina.akbari@asu.edu

Adolfo R. Escobedo
School of Computing and Augmented Intelligence
Arizona State University
Tempe, Arizona, United States
adres@asu.edu

***Abstract*—Rank aggregation has many applications in computer science, operations research, and group decision-making. This paper introduces lower bounds on the Kemeny aggregation problem when the input rankings are non-strict (with and without ties). It generalizes some of the existing lower bounds for strict rankings to the case of non-strict rankings, and it proposes shortcuts for reducing the run time of these techniques. More specifically, we use Condorcet criterion variations and the Branch & Cut method to accelerate the lower bounding process.**

***Index Terms*—Rank aggregation, lower bounding techniques, group decision-making, Kemeny-Snell distance, Condorcet Criterion, branch & cut method.**

I. Introduction

The rank aggregation problem is utilized in situations when it is necessary to reconcile the evaluations of m judges who rank n alternatives according to some criteria. Rank aggregation is often used to guide group decision-making. In particular, it is useful in decision support systems to find a robust collective decision instead of relying on individuals. Rank aggregation has numerous applications such as recommender systems [1], [2], feature selection [3], [4], and bioinformatics [5], [6]. The purpose of rank aggregation problem is to find a ranking that either maximizes a scoring function or minimizes a distance measure to the input rankings; the former are known as score-based methods, and the latter as distance-based methods. Examples of score-based methods are Borda rule [7] and Copeland rule [8]; examples of distance-based methods are Kemeny aggregation [9] and Spearman's footrule [10]. It is important to remark that Kemeny aggregation, which uses the Kemeny-Snell distance [9], is one of the most widely used rank aggregation methods. This popularity comes from the fact that this measure uniquely satisfies a set of axioms, namely anonymity, commutativity, extension, non-negativity, scaling, and triangular inequality [9]. Additionally, it satisfies a set of key social choice properties including neutrality, local stability, and the Condorcet criterion [11], [12].

Kemeny aggregation is NP-hard even when there are only four input rankings [10], [13]; however, there have been numerous efforts to solve this problem to optimality. The vast majority of exact methods can only handle strict rankings such as [14], [15]; a small number are applicable to both strict and non-strict rankings [16]–[18]. Since Kemeny aggregation is NP-hard, many heuristic methods [19]–[21] and approximation algorithms [10], [22], [23] have been proposed in the literature. For more examples, we refer the reader to [24].

Some researchers have studied partitioning methods that leverage the fact that the optimal solutions to Kemeny aggregation satisfy certain social choice properties. Using these partitioning schemes, the original problem can be decomposed into smaller subproblems where solving the subproblems independently and concatenating the results is guaranteed to reproduce the optimal solution to the original problem. One such scheme is the Extended Condorcet Criterion (XCC) [25]. Yoo and Escobedo [18] showed that the solution to Kemeny aggregation with non-strict rankings may violate XCC; thus, the authors proposed the Non-strict Extended Condorcet Criterion (NXCC), which is suitable for both strict and non-strict rankings. For the rest of the paper, we denote these partitioning methods as *Condorcet-based partitioning* for simplicity. These partitioning schemes can render Kemeny aggregation easier to solve [18], [26]. Herein, we use these partitioning schemes to facilitate the proposed lower bounding techniques. We remark that here are other partitioning schemes such as *3/4-Majority Rule* [26], which is defined for strict rankings; however, XCC partitioning is always as good as partitioning using the 3/4-Majority Rule.

Obtaining high quality lower bounds is of high importance, as these bounds are usually used to evaluate the quality of heuristics and approximation algorithms. More specifically, a low quality lower bound may lead to the incorrect conclusion that an optimal or near-optimal solution is of low quality. There are three general classes of lower bounding techniques for Kemeny aggregation: 1) pairwise comparison methods, 2) cycle-based methods, and 3) LP relaxation methods. Pairwise comparison methods leverage the fact that each pair of alternatives contributes a minimum amount to the overall distance. Cycle-based methods seek to improve pairwise comparison lower bounding techniques by taking advantage of the fact that the preferences returned by the solution must be transitive. Finally, a lower bound can be obtained by solving the LP relaxation of the Kemeny aggregation formulation. We remark that there are other infrequently used ways to obtain a lower bound on Kemeny aggregation with strict rankings such as using Spearman's footrule [10] and Borda count [27], based

978-1-7281-9048-8/21/$31.00 ©2021 IEEE

on their relationship with the Kemeny-Snell distance; however, to the best of our knowledge, these relationships have not been extended to the case of non-strict rankings.

This work generalizes the three aforementioned classes of lower bounding techniques for Kemeny aggregation with strict rankings to the case with non-strict rankings. As another contribution, it accelerates the run time of lower bounding techniques using the Branch & Cut (B&C) method and Condorcet-based partitioning. These techniques have been largely used for solving exact formulation of Kemeny aggregation. Using B&C, it is possible to reduce the run time of LP relaxation methods. Furthermore, with the help of XCC and NXCC, the process of preference-cycle detection and the solution of LP relaxation-based methods can be significantly expedited (with and without B&C) for certain instances.

The rest of the paper is organized as follows. Section II introduces the notation used throughout the paper, and it provides preliminary definitions. Section III reviews the lower bounds obtained from pairwise comparison information for strict rankings and generalizes this method for non-strict rankings. Section IV focuses on cycle-based lower bounding techniques, and it leverages Condorcet-based partitioning for the purpose of accelerating cycle-based methods. Section V focuses on LP relaxation lower bounding techniques. Section VI describes the experiments and discusses the results. Finally, Section VII concludes the paper.

II. Notation and Preliminaries

Rankings can be categorized as strict and non-strict. Strict rankings refer to the case where there are no ties, while non-strict rankings refer to the case where there may be ties. Both strict and non-strict rankings can further be categorized into complete and incomplete, e.g. see [10], [28]; all alternatives are ranked in the former but some alternatives may be unranked in the latter. This work focuses on complete and possibly non-strict rankings.

Let $\mathcal{A} = \{1, 2, \cdots, n\}$ be the set of alternatives, and $\mathcal{L} = \{1, 2, \cdots, m\}$ be the set of input rankings, i.e., the judges' ordinal evaluations. Additionally, let $\boldsymbol{\Sigma} \subset \mathbb{Z}^n$ be the set of all possible complete rankings, $\boldsymbol{\sigma}^l \in \boldsymbol{\Sigma}$ be the ranking of judge $l \in \mathcal{L}$, $\sigma^l(i)$ be the rank of alternative i in the ranking of judge l, and $\boldsymbol{\sigma}^*$ be the consensus ranking. As a convention, $\sigma^l(i) < \sigma^l(j)$ indicates that alternative i is preferred over alternative j by judge l, and $\sigma^l(i) = \sigma^l(j)$ indicates that alternatives i and j are tied by judge l. Furthermore, let $\Lambda = \{(i,j)|i,j \in \mathcal{A},\ j > i\}$ be the set of unordered pairs of distinct alternatives.

The Kemeny-Snell distance between two complete rankings $\boldsymbol{\sigma}^1$ and $\boldsymbol{\sigma}^2$, denoted by $D_{ks}(\boldsymbol{\sigma}^1, \boldsymbol{\sigma}^2)$, is given by

$$D_{ks}(\boldsymbol{\sigma}^1, \boldsymbol{\sigma}^2) = \frac{1}{4} \sum_{i,j \in \mathcal{A}} |\text{sign}(\sigma^1(i) - \sigma^1(j)) - \text{sign}(\sigma^2(i) - \sigma^2(j))|. \tag{1}$$

Note that $sign(v)$ returns 1 if $v > 0$, -1 if $v < 0$, and 0 otherwise. In the case of strict rankings, D_{ks} counts the number of rank reversals between two given rankings $\boldsymbol{\sigma^1}$ and $\boldsymbol{\sigma^2}$. Furthermore, in the case of non-strict rankings, D_{ks} assigns half of a rank reversal whenever alternatives i and j are tied in one ranking, but not in the other.

Kemeny aggregation can be mathematically stated as:

$$\boldsymbol{\sigma}^* = \underset{\boldsymbol{\sigma} \in \boldsymbol{\Sigma}}{\text{argmin}} \sum_{l \in \mathcal{L}} D_{ks}(\boldsymbol{\sigma}, \boldsymbol{\sigma^l}). \tag{2}$$

As a convention, let D_{ks}^* be the cumulative Kemeny-Snell distance between $\boldsymbol{\sigma}^*$ and all of the input rankings. The focus of this paper is finding lower bounds on D_{ks}^*.

Let $p_{ij} = |l \in \mathcal{L} : \sigma^l(i) < \sigma^l(j)|$ and $t_{ij} = |l \in \mathcal{L} : \sigma^l(i) = \sigma^l(j)|$ be the number of input rankings in which alternative i is ranked ahead of alternative j and the number of input rankings in which alternatives i and j are tied, respectively.

Alternative i is said to be *pairwise preferred by a decisive majority* over alternative j, denoted by $i \succ j$, if $p_{ij} > p_{ji} + t_{ij}$ in the case of non-strict rankings [18]. If neither i is preferred over j nor j is preferred over i, then there is no decisive majority preference between i and j; this case is denoted by $i \approx j$. To be more succinct, for the rest of the paper we use the term *pairwise preferred* instead of *pairwise preferred by a decisive majority*, for succinctness.

Yoo and Escobedo [18] proposed the Generalized Kemeny Binary Programming (GKBP) formulation for solving Kemeny aggregation with strict and non-strict rankings. GKBP utilizes the relationship between Kemeny-Snell distance and the extended Kendall tau correlation coefficient [17]. The formulation is given by:

$$\max \quad z = \sum_{i,j \in \mathcal{A}} c_{ij}(2y_{ij} - 1) \tag{3a}$$

$$\text{s.t. } y_{ij} - y_{kj} - y_{ik} \geq -1,\ i,j,k \in \mathcal{A};\ i \neq j \neq k \tag{3b}$$

$$y_{ij} + y_{ji} \geq 1, \quad i,j, \in \mathcal{A};\ i \neq j \tag{3c}$$

$$y_{ii} = 0, \quad i \in \mathcal{A}, \tag{3d}$$

$$y_{ij} \in \{0, 1\}, \quad i,j \in \mathcal{A};\ i \neq j. \tag{3e}$$

Here, decision variable y_{ij} is equal to 1 if alternative i is ranked ahead or tied with alternative j and 0 otherwise; i and j are tied in the ranking if $y_{ij} = y_{ji} = 1$. Objective function (3a) maximizes the extended Kendall tau correlation coefficient; the coefficient c_{ij} is given by $p_{ij} + t_{ij} - p_{ji}\ \forall i,j \in \mathcal{A}$ when the input rankings are complete. Constraint (3b) prevents preference cycles. Constraint (3c) enforces that i and j cannot be simultaneously dispreferred over each other. Constraint (3e) determines the domain of the variables. Let $\boldsymbol{\sigma'}$ be the ranking obtained by solving GKBP, whose ith element (i.e., the rank of alternative i) is calculated as $\sigma'(i) = n - \sum_{j \neq i} y_{ij}$. Furthermore, the cumulative Kemeny-Snell distance between $\boldsymbol{\sigma'}$ and all of the input rankings, $\sum_{l \in \mathcal{L}} D_{ks}(\boldsymbol{\sigma'}, \boldsymbol{\sigma^l})$, is calculated as $(mn(n-1) - z)/2$, where z is the objective function value of GKBP. Formulation 3 is selected for solving the LP relaxation of Kemeny aggregation with non-strict rankings since [18] showed that it is more computationally advantageous than the binary programming formulation proposed in [16].

III. Pairwise Comparison Methods

The cumulative Kemeny-Snell distance between any $\boldsymbol{\sigma} \in \boldsymbol{\Sigma}$ and all of the input rankings, $\sum_{l \in \mathcal{L}} D_{ks}(\boldsymbol{\sigma}, \boldsymbol{\sigma}^l)$, can be expressed as $\sum\limits_{(i,j)\in\mathcal{A}} D_{ks}(\sigma_{ij})$, where $D_{ks}(\sigma_{ij})$ is the contribution of each pair of alternatives $(i,j) \in \mathcal{A}$ to $\sum_{l\in\mathcal{L}} D_{ks}(\boldsymbol{\sigma}, \boldsymbol{\sigma}^l)$.

For the case of strict rankings, $D_{ks}(\sigma_{ij})$ is given by

$$D_{ks}(\sigma_{ij}) = \begin{cases} 2p_{ji} & \text{if } \sigma(i) < \sigma(j), \\ 2p_{ij} & \text{if } \sigma(j) < \sigma(i). \end{cases}$$

Davenport and Kalagnanam [29] proposed the first lower bound for Kemeny aggregation with strict ranking, given by

$$LB_0 = 2 \sum_{(i,j)\in\mathcal{A}} \min(p_{ij},\, p_{ji}). \tag{4}$$

Eq. (4) obtains a lower bound on D^*_{ks} by simply summing the smallest contribution of all distinct pairs of alternatives. Note that Eq. (4) has been scaled by a factor of 2 herein to facilitate the generalization of this lower bound to non-strict rankings.

For the case of non-strict rankings, $D_{ks}(\sigma_{ij})$ can be expressed as [18]

$$D_{ks}(\sigma_{ij}) = \begin{cases} 2p_{ji} + t_{ij} & \text{if } \sigma(i) < \sigma(j), \\ 2p_{ij} + t_{ij} & \text{if } \sigma(j) < \sigma(i), \\ p_{ij} + p_{ji} & \text{if } \sigma(i) = \sigma(j). \end{cases} \tag{5}$$

Eq. (5) can be intuitively described as follows. While computing the distance between $\boldsymbol{\sigma}$ and input ranking $\boldsymbol{\sigma}^l$, D_{ks} imposes a distance of 2 if the relative ordering of alternatives i and i is different in the two rankings; a distance of 1 if i and i are tied in one ranking but not in the other; and a distance of 0 when both rankings concur on the relative ordering of i and j. Therefore, if $\sigma(i) < \sigma(j)$, Eq. (5) is equal to the number of input rankings in which j is ranked ahead of i, times 2, plus the number of input rankings in which i and j are tied. Additionally, if $\sigma(j) = \sigma(i)$, Eq. (5) is equal to the number of input rankings in which i and j are not tied. Using this information, we generalize LB_0 to the case of non-strict rankings.

Proposition 1. *Given an instance of Kemeny aggregation, a lower bound on D^*_{ks} is given by*

$$LB_1 = \sum_{(i,j)\in\mathcal{A}} \min(2p_{ij} + t_{ij},\, 2p_{ji} + t_{ij},\, p_{ij} + p_{ji}). \tag{6}$$

Similar to LB_0, Eq. (6) obtains a lower bound on D^*_{ks} by simply summing the smallest contribution of all distinct pairs of alternatives. Note that LB_1 reduces to LB_0 when the input rankings are strict, i.e., if $t_{ij} = 0\ \forall i, j \in \mathcal{A}$.

IV. Cycle-Based Methods

This section is devoted to lower bounds on Kemeny aggregation using preference-cycles. First, Section IV-A reviews cycle-based methods for strict rankings; Section IV-B proposes a method to apply these techniques to the case of non-strict rankings; and Section IV-C uses Condorcet-based partitioning to reduce the run time of cycle-based methods.

A. Cycle-Based Methods for Strict Rankings

The Kemeny aggregation problem with strict rankings can be solved via the Weighted Minimum Feedback Arc Set Problem (WMFASP), and vice versa [14], [30]. Let $G = (V, E)$ be a weighted directed graph where V is the set of vertices and E is the set of arcs (edges). The objective of WMFASP is finding a subset of arcs $E' \subset E$ with minimum weight such that its removal would make the resulting graph, i.e., $G' = (V, E \backslash E')$, acyclic [31]. Conitzer, Davenport, and Kalagnanam [14] provided various lower bounds for the equivalent WMFASP of Kemeny aggregation with strict rankings on the *pairwise majority graph*. The nodes of this graph are the alternatives; there is a directed arc from i to j if $p_{ij} > p_{ji}$ with a weight of $w_{ij} = p_{ij} - p_{ji}$; and there is no arc from i to j and vice versa if $p_{ij} = p_{ji}$. The lower bounds on the WMFASP pairwise majority graph in [14] do not provide any information on how to obtain the respective lower bounds on the equivalent Kemeny aggregation problem. For this reason, Milosz and Hamel [32] utilized methods developed in [14] to improve LB_0.

Recall that LB_0 considers the smallest contribution for each pair of alternatives. However, the resulting solution obtained by this selection may not be transitive as it may contain preference-cycles, which can make this bound unattainable. Nevertheless, this information can be utilized to improve this lower bound.

Let $\overline{C} = \{c_1, c_2, \ldots, c_s\}$ be any set of edge-disjoint preference-cycles. For each cycle, the consensus ranking disagrees with at least one of the edges in the cycle [14]. Hence, the lower bound can be improved by adding the minimum cost of reverting an edge of cycle, i.e., the minimum cost of breaking the preference-cycle. Therefore, a new lower bound can be calculated as follows [14], [32]:

$$LB_2 = LB_0 + \sum_{c_r\in\overline{C}} \min_{(i,j)\in c_r} w_{ij}. \tag{7}$$

When only edge-disjoint preference-cycles are considered, a part of the cycles remains unused. Conitzer, Davenport, and Kalagnanam [14] proposed a method to leverage this underutilized information. Let $C = \{c_1, c_2, \ldots, c_s\}$ be any set of preference-cycles, and $\delta\left((i,j), c_r\right)$ be an indicator function which is set to 1 if $(i,j) \in c_r$, and 0 otherwise. Additionally, let $v_l = \min_{(i,j)\in c_r}\{w_{ij} - \sum_{q=1}^{l-1} \delta\left((i,j), c_q\right).v_q\}$ [14]. Intuitively, v_l calculates the minimum portion of the weights of c_r edges that have not been used by prior cycles in C. A lower bound on the Kemeny aggregation problem with strict rankings can be calculated as [14], [32]

$$LB_3 = LB_0 + \sum_{c_r\in C} v_l. \tag{8}$$

Notice that LB_3 is at least as good as LB_2 [14].

B. Cycle-Based Methods for Non-Strict Rankings

LB_1 provides a lower bound on Kemeny aggregation with non-strict rankings using pairwise comparison information by considering the smallest among all three possible values.

Similar to the case of strict rankings, the resulting ranking of this selection may contain preference-cycles, which can be similarly broken to boost LB_1.

Similar to [14] and [32], we focus only on preference-cycles of length 3 for the purposes of simplicity and computational efficiency. These preference-cycles are much easier to find, and every preference-cycle of length 4 or higher contains at least a preference-cycle of length 3 [33].

As mentioned earlier, Kemeny aggregation with strict rankings and WMFASP are equivalent problems. However, this claim has not yet been proven for non-strict rankings. We reckon that, to the best of our knowledge, it may not be possible to represent Kemeny aggregation with non-strict rankings via an equivalent WMFASP. The reason is that, for every pair of distinct alternatives (i, j), there are 3 parameters involved, namely p_{ij}, p_{ji}, t_{ij}. Additionally, in the case of strict rankings, alternatives i, j, k form a preference-cycle if $i \succ j \succ k \succ i$; however, in the case of non-strict rankings, there are additional types of preference cycles as shown in Fig. 1, where arc (i, j) is drawn if $i \succ j$; and arcs (i, j) and (j, i) are simultaneously drawn if $i \approx j$. In the case of strict rankings, it is possible to break a preference-cycle of length 3 by reversing the edge with the lowest weight, however, this method cannot be applied to non-strict rankings, as edges are not weighted; additionally, reversing certain individual edges may not break the cycle. For example, reversing edge $(k,\ j)$ in Fig. 1 (b) does not make the resulting graph acyclic. Consequently, it is not possible to directly apply the previously reviewed techniques for strict rankings to the case of non-strict rankings.

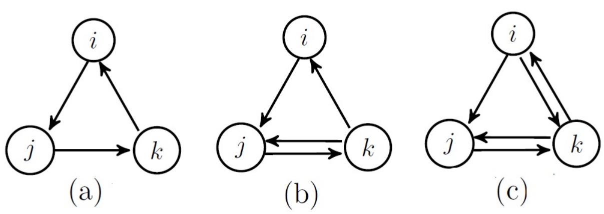

Figure 1. preference-cycles in non-strict rankings [18]

We propose a new method to boost LB_1 using preference-cycles. Let $\overline{C} = \{c_1, \dots, c_s\}$ be any set of edge-disjoint preference-cycles of length 3. For each cycle $c_r \in \overline{C}$, we explicitly evaluate all 13 possible non-strict rankings (i.e., acyclic preferences) of 3 alternatives, and we define X_r^* as the minimum D_{ks} distance of the Kemeny aggregation restricted to the alternatives in cycle c_r. X_r^* is the minimum possible contribution of alternatives i, j, k in D_{ks}^*, i.e., consensus ranking. On the other hand, the contribution of pairs $(i, j), (i, k)$, and (j, k) to LB_1, denoted by d_{ijk}, is equal to

$$d_{ijk} = 2\left[\min D_{ks}(\sigma_{ij}) + \min D_{ks}(\sigma_{ik}) + \min D_{ks}(\sigma_{jk})\right]. \tag{9}$$

We remark that Eq. (9) has been multiplied by 2 since D_{ks} counts each pair of alternatives twice.

Hence, the improvement caused by breaking cycle c_r, denoted by Q_r, is equal to

$$Q_r = X_r^* - d_{ijk}.$$

As a result, an improved lower bound can be obtained as

$$LB_4 = LB_1 + \sum_{c_r \in \overline{C}} Q_r. \tag{10}$$

Given a set of cycles, it is possible to construct different edge-disjoint sets that can result in different values of LB_4. Since LB_1 is a fixed value, LB_4 is maximized by focusing on the second term of Eq. (10).

The problem of finding the set of disjoint cycles that yields the highest LB_4 value can be formulated as a weighted node packing (WNP) problem. Let $G = (V, E, W)$ be an undirected graph where V is the set of nodes, E is the set of edges, and W is set of nodes weights. The goal of WNP is finding a subset of nodes with maximum total weight such that no pair of nodes shares an edge [34]. Here, V is the set of cycles, i.e., $V = C$. There is an edge between cycles $c_r, c_g \in C$ if c_r and c_g are not edge-disjoint. The weight of node c_r is its improvement to the lower bound, i.e., Q_r.

Let $C = \{c_1, \dots, c_s\}$ be the set of all cycles of length 3, which may not necessary be edge-disjoint. Given C, the ensuing optimization problem maximizes LB_4. Beforehand, we introduce the decision variables and parameters of the model. Let decision variable v_l be equal to 1 if cycle $c_r \in C$ is in the selected set of edge-disjoint cycles, and 0 otherwise. Additionally, let Ξ be the set of cycle pairs that share an edge. The binary programming formulation is given by

$$\max \quad \sum_{c_r \in C} Q_r v_r \tag{11a}$$

$$\text{subject to} \quad v_r + v_g \leq 1, \quad \forall (c_r, c_g) \in \Xi \tag{11b}$$

$$v_r \in \{0, 1\}, \quad \forall c_r \in C. \tag{11c}$$

Objective function (11a) maximizes LB_4; Constraint (11b) enforces that, whenever cycles c_r and c_g share one edge, at most one of them can belong to the set of edge-disjoint cycles; and Constraint (11c) specifies the domain of the decision variables. WNP is an NP-hard problem for general graphs and even finding an approximation algorithm for this problem is NP-hard [34]. WNP has stronger formulations using cliques [34]. However, solving this problem to optimality still may be computationally demanding.

Here, we propose a simple add-swap heuristic method to find a high quality set of edge-disjoint cycles for our problem of interest. The pseudocode of the proposed method is presented in Algorithm 1, which begins by sorting cycles based on non-increasing improvement in the lower bound, i.e., Q-values. Additionally, let C' be the working set of edge-disjoint cycles, which initially is set to be empty. Next, starting with the cycle that yields the highest improvement, the algorithm adds this cycle to C' if its addition keeps the working set edge-disjoint. After this step, the algorithm checks whether it

Algorithm 1: Lower Bound Improvement using preference-cycles

Input : Set of cycles of length 3 (C), $Q = [Q_r] \in \mathbb{Z}^{|C|}$
Output : Overall improvement in the lower bound
Sort cycles based on non-increasing Q-values;
Discard cycles with zero improvement;
$C' \leftarrow \emptyset$; `// Set of edge disjoint cycles`
$\overline{Q} \leftarrow 0$; `// Overall improvement in the lower bound`
for $l = 1$ **to** $|C|$ **do**
  if *Cycle* c_r *is mutually edge-disjoint with all cycles in* C' **then**
    $C' \leftarrow C' \cup c_r$;
    $\overline{Q} \leftarrow \overline{Q} + Q_r$
while *True* **do**
  Swap cycles $c_m, c_n \notin C'$ with $c_g \in C'$ if this swap allows the set to remain edge-disjoint and increases $\overline{Q}$;
  Otherwise, set to False
return $\overline{Q}$

is possible to swap one cycle in C' with two mutually edge-disjoint cycles that are not in C', such that this swap keeps C' to remain edge-disjoint and simultaneously increases the overall lower bound improvement of C'.

Detecting preference-cycles of length 3 has a time complexity of $O(n^3)$, which makes this lower bound boosting technique suitable only for small to medium sized problem. Here, we use Condorcet-based partitioning to reduce the run time of this process.

C. Scaling up Cycle-Based Methods with Social Choice Properties

The Condorcet Criterion [35] is one of the most recognized and widely utilized social choice properties. Young [36] formalized this concept, which states that if there exists an alternative such that it beats all other candidates in pairwise comparison, then it must ranked first in all consensus rankings. Such an alternative is known as the *Condorcet Winner*. The Condorcet Criterion can be mathematically stated as

$$\text{if } \exists\, i \in \mathcal{A} : \; p_{ij} > p_{ji} \; \forall j \in \mathcal{A} \backslash i \implies \; \sigma^*(i) < \sigma^*(j) \;\; \forall j \in \mathcal{A} \backslash i,$$

where $\boldsymbol{\sigma}^*$ is consensus ranking. Truchon [25] proposed a more general version of the Condorcet Criterion, called the Extended Condorcet Criterion (XCC), which can be applied to subsets of alternatives rather than to a single alternative. XCC guarantees the relative ordering of alternatives in different subsets in the consensus rankings. Let $\boldsymbol{\mathcal{A}} = \{\mathcal{A}_1, \mathcal{A}_2, \cdots, \mathcal{A}_w\}$ be a partition of X, where all alternatives in lower-indexed subsets are pairwise preferred over all alternatives in higher-indexed subsets. XCC guarantees that all alternatives in lower-indexed subsets are ranked ahead of all alternatives in the higher-indexed subsets in the consensus ranking. This criterion can be mathematically stated as

$$\begin{aligned} &\text{if } p_{ij} > p_{ji} \; \forall i \in X_k \in \boldsymbol{\mathcal{A}}, \forall j \in \mathcal{A}_{k'} \in \boldsymbol{\mathcal{A}} : k < k' \\ &\quad \implies \sigma^*(i) < \sigma^*(j) \; \forall i \in \mathcal{A}_k \in \boldsymbol{\mathcal{A}}, \forall j \in \mathcal{A}_{k'} \in \boldsymbol{\mathcal{A}}. \end{aligned}$$

Recently, Yoo and Escobedo [18] showed that XCC may not be consistent with the optimal solution of Kemeny aggregation with non-strict rankings; hence the authors proposed the Non-strict Extended Condorcet Criterion (NXCC), which is consistent with the solution to the aforementioned problem. NXCC can be mathematically stated as:

$$\begin{aligned} &\text{if } p_{ij} > p_{ji} + t_{ij} \; \forall i \in \mathcal{A}_k \in \boldsymbol{\mathcal{A}}, \forall j \in \mathcal{A}_{k'} \in \boldsymbol{\mathcal{A}} : k < k' \\ &\quad \implies \sigma^*(i) < \sigma^*(j), \; \forall i \in \mathcal{A}_k \in \boldsymbol{\mathcal{A}}, \forall j \in \mathcal{A}_{k'} \in \boldsymbol{\mathcal{A}}. \end{aligned}$$

It is important to remark that NXCC is a generalization of XCC, as it becomes XCC when all input rankings are strict.

Since applying NXCC has a time complexity of $O(n^2)$ [18], [25], this modification can make cycle detection operation less expensive and may be effective for certain large-scale problems. Let $\boldsymbol{\mathcal{A}} = \{\mathcal{A}_1, \mathcal{A}_2, \cdots, \mathcal{A}_w\}$ be the partition according to NXCC. In the original problem, all $\binom{n}{3}$ alternative triplets must be checked for possible preference cycles. However, in the partitioned problem it is sufficient to evaluate all subsets of $\boldsymbol{\mathcal{A}}$ for possible preference cycles independently. Hence, in the partitioned problem only $\binom{|\mathcal{A}_1|}{3} + \binom{|\mathcal{A}_2|}{3} + \ldots + \binom{|\mathcal{A}_w|}{3}$ triplets must be checked.

Proposition 2. *Given an instance of Kemeny aggregation, the set of cycles obtained from the NXCC partition and from the original non-partitioned problem are the same.*

Proof. All the alternatives that form a preference-cycle must belong to the same subset in the NXCC partition [25]. Therefore, there are no preference cycles between alternatives from different subsets. □

V. LP Relaxation-Based Methods

It is well known that the LP relaxation version of a minimization Integer Programming (IP) model provides a valid lower bound on the respective IP model. Conitzer, Davenport, and Kalagnanam [14] explored this type of lower bound on the WMFASP version of the Kemeny aggregation with strict rankings. This method yields tighter bounds than cycle-based methods; however, it takes more time as well. This is not surprising, since the exact formulation of the Kemeny aggregation model, and hence the LP relaxation version, has $O(n^3)$ constraints, which can be cumbersome to solve for large values of n. However, a large portion of these constraints are trivially satisfied at the optimal solution [37], which is a fact that can be utilized to simplify the solution to the LP relaxation models of Kemeny aggregation. Here, we explore an alternative exact solution approach for solving the LP relaxation problem, the branch and cut (B&C) method [37]. B&C is an iterative optimization approach that begins with a relaxed version of a problem's exact formulation; the relaxation usually excludes a large number of the constraints and is, hence, easier to solve. Here, the cycle-prevention constraints are excluded from the model at first, i.e, Constraints (3b). At each iteration, the working relaxation is solved to optimality and the solution is analyzed to determine if any of the excluded constraints are violated, in which case the respective constraints (i.e., cuts) are added back into the working relaxation model. This process is repeated until there are no such violations.

Furthermore, the process of obtaining a lower bound using the LP relaxation method can be accelerated by deploying NXCC partitioning. If the instance of interest is partitionable, the original problem can be equivalently solved as a collection of smaller subproblems, which can decrease run times.

VI. Computational Results

This section compares the quality and run time of the various lower bounding techniques for non-strict rankings discussed in this work. Condorcet-based partitioning was performed using the algorithm proposed in [18], which works by carrying out sequential pairwise comparisons. All experiments were carried out on a computer with an Intel(R) Xeon(R) CPU E5-2680 @ 2.40 GHz with 64 GB RAM. The optimization models were solved using CPLEX solver version 12.10.0.

A. Data Set

We use the Mallows model [38], which is a popular probabilistic model on ranking data and has the nice property of scalability [39] to generate synthetic instances. Using the Mallows model, we can control the difficulty of generated instances and investigate the performance of the algorithms under different conditions.

The Mallows model has two parameters: a ground truth ranking $\overline{\boldsymbol{\sigma}}$, and dispersion $\phi \in (0, 1]$. The Mallows model used in this work utilizes Kemeny-Snell distance. The dispersion parameter controls the density of the generated ranking around $\overline{\boldsymbol{\sigma}}$. The probability of observing a ranking $\boldsymbol{\sigma}$ is determined by:

$$P(\boldsymbol{\sigma}) = P(\boldsymbol{\sigma}|\overline{\boldsymbol{\sigma}}, \phi) = \frac{\phi^{D_{ks}(\boldsymbol{\sigma}, \overline{\boldsymbol{\sigma}})}}{Z}, \tag{12}$$

where $Z = \sum_{\boldsymbol{\sigma'} \in \boldsymbol{\Sigma}} \phi^{D_{ks}(\boldsymbol{\sigma'}, \overline{\boldsymbol{\sigma}})} = 1 \times (1 + \phi) \times (1 + \phi + \phi^2) \times \cdots \times (1 + \cdots + \phi^n - 1)$ is a normalizing constant. When ϕ approaches 0, the Mallows model generates a ranking closer to $\overline{\boldsymbol{\sigma}}$ and, as ϕ approaches 1, Eq. (12) converges to a uniform distribution, which means any complete ranking has an equal probability of occurring.

Doignon et al. [40] introduced the Repeated Insertion Model (RIM) for generating strict rankings, which encompasses the Mallows model as a special case. To describe RIM, assume without loss of generality, that the ground truth ranking $\overline{\boldsymbol{\sigma}}$ is the permutation $(1, 2, \ldots, n)$. The method starts by placing alternative 1 into an initially empty working ranking vector; in each succeeding iteration and until the target size is reached, the next alternative from $\overline{\boldsymbol{\sigma}}$ is inserted in a specific position in the working ranking vector based on the Mallows probabilities. Specifically, alternative i is inserted before alternative $j < i$ in the working ranking vector with probability $p_{ij} = \phi^{i-j}/(1 + \phi + \cdots + \phi^{i-1})$. Yoo and Escobedo [41] developed a modified RIM sampling process for generating non-strict rankings, which is used herein. In this sampling process, after generating strict rankings via RIM, a random number u is drawn from a uniform distribution $U(1, n-1)$, and the alternative with rank u is tied with the alternative with the next higher (i.e., worse) rank. The process is repeated until the number of alternatives that are tied reaches a specific threshold, herein set to $0.25n$. Please refer to [40] and [41] for more information.

The tested parameter settings are $\phi \in \{0.8, 0.85, 0.9, 0.95\}$, $n \in \{50, 100, 150, 200\}$, and $m = 20$; we chose only high values of ϕ because they are more difficult to solve, as they correspond to low group cohesion and higher noise levels [41].

B. Results and Discussion

For each combination of (ϕ, n), we perform 20 replications. Since the ground truth ranking used in the Mallows model are the same for each combination of (ϕ, n), the D^*_{ks} values very close to each other. For all three tested lower bounding techniques, the experimental results shown in Table I reports the average, minimum, and maximum lower bound over the 20 replications for each combination of (ϕ, n). Furthermore, Table I reports the geometric mean run time of the pairwise comparison (PC); the run time of the cycle-based method (CB) and its run time with NXCC (CB + NXCC); and the run time of the LP relaxation (LPR), its run time with B&C (LPR + B&C), and its run time with B&C and NXCC (LPR + B&C + NXCC) over the 20 replications for each combination of (ϕ, n).

Instances with a lower ϕ value yielded a partition with more subsets than those with a higher value. This was expected since higher values of ϕ correspond to more noise in the generated rankings, which induces less agreement on the relative ordering of alternatives in these instances. All tested instances with a ϕ value of 0.8, 0.85, 0.9 yielded a non-trivial NXCC partition, however, all tested instances with a ϕ value of 0.95 were not partitionable. Whenever, NXCC yielded a non-trivial partition, it was able to accelerate the preference-cycle detection process rather significantly, especially for instances with a ϕ value of 0.8 and 0.85 where the NXCC partition had the most subsets.

For all tested combinations of (ϕ, n), the LP relaxation method achieves a better average, minimum, and maximum lower bound than the other two techniques; furthermore, this was true for all the individual instances. On the other hand, LP relaxation had the highest average run time for all tested combinations of (ϕ, n); this was true for all the individual instances as well. As expected, the cycled-based technique achieves a better lower bound than the pairwise comparison method; on the other hand, it takes significantly more time. The pairwise comparison method achieves the worst bounds. However, the highest run time of this technique was only 0.04 seconds, which makes this method very attractive whenever a very fast lower bounding technique is required.

The cycle-based technique was able to improve the pairwise comparison lower bound in up to a handful of seconds. In fact this technique was able to achieve bounds that were competitive with the LP relaxation technique in far less time. Deploying NXCC makes this method even more attractive as, whenever NXCC yielded a non-trivial partition, it was able to accelerate the cycle-based technique rather impressively,

Table I
COMPUTATIONAL RESULTS OF DIFFERENT LOWER BOUNDING TECHNIQUES

N=50															
ϕ	Pairwise Comparison (PC)				Cycle-Based (CB)					LP Relaxation (LPR)					
	LB			Time	LB			Time		LB			Time		
	Ave	Min	Max	PC	Ave	Min	Max	CB	CB + NXCC	Ave	Min	Max	LPR	LPR + B&C	LPR + B&C + NXCC
0.8	6490.25	5974	6892	0.00	6510.35	6004	6906	0.06	0.01	6513.05	6006	6906	4.20	1.68	0.22
0.85	8651.65	7940	9418	0.00	8684.85	7956	9452	0.06	0.03	8689.7	7956	9458	4.27	1.84	0.41
0.9	11782.75	10726	12854	0.00	11842.35	10796	11806.5	0.09	0.10	11852.55	10810	12940	4.24	1.94	1.20
0.95	16705.15	15633	17906	0.00	16856.75	15777	16790.5	0.37	0.39	16892.55	15811	16817.5	4.35	1.93	1.86

N=100															
ϕ	Pairwise Comparison (PC)				Cycle-Based (CB)					LP Relaxation (LPR)					
	LB			Time	LB			Time		LB			Time		
	Ave	Min	Max	PC	Ave	Min	Max	CB	CB + NXCC	Ave	Min	Max	LPR	LPR + B&C	LPR + B&C + NXCC
0.8	14263.8	13585	15073	0.01	14304.8	13615	15109	0.38	0.03	14311.7	13617	15117	35.8	15.20	1.34
0.85	19654.55	18644	20615	0.01	19725.55	18724	20701	0.39	0.08	19738.95	18738	20727	35.63	15.57	1.62
0.9	29167.35	27445	30625	0.01	29308.45	27583	30799	0.53	0.51	29335.85	27623	30831	35.56	15.98	10.01
0.95	49735.15	46784	52936	0.01	50059.35	47030	53310	2.16	2.19	50137.35	47091	53392	36.11	15.38	15.41

N=150															
ϕ	Pairwise Comparison (PC)				Cycle-Based (CB)					LP Relaxation (LPR)					
	LB			Time	LB			Time		LB			Time		
	Ave	Min	Max	PC	Ave	Min	Max	CB	CB + NXCC	Ave	Min	Max	LPR	LPR + B&C	LPR + B&C + NXCC
0.8	22276.6	21578	23189	0.02	22345.3	21642	23273	1.26	0.08	22357.1	21652	23291	122.32	52.67	3.43
0.85	30640.1	29563	31850	0.02	30745.6	29669	31974	1.26	0.18	30766.75	29692	32004	122.30	56.02	3.96
0.9	47121.75	44486	49368	0.02	47324.25	44686	49624	1.49	1.46	47364.05	44718	49682	121.75	55.65	31.92
0.95	85673.6	84089	86765	0.02	86168	84577	87315	6.75	6.76	86282	84713	87451	121.06	58.80	58.81

N=200															
ϕ	Pairwise Comparison (PC)				Cycle-Based (CB)					LP Relaxation (LPR)					
	LB			Time	LB			Time		LB			Time		
	Ave	Min	Max	PC	Ave	Min	Max	CB	CB + NXCC	Ave	Min	Max	LPR	LPR + B&C	LPR + B&C + NXCC
0.8	29850.1	29046	30894	0.03	29935.1	29118	30980	2.79	0.11	29948.1	29120	31002	294.98	131.80	8.28
0.85	41825.2	40709	43203	0.03	41969.4	40855	43339	2.86	0.28	41992	40879	43363	296.57	135.50	6.71
0.9	64462.8	62991	68004	0.04	64742.6	63239	68290	3.40	2.84	64792.7	63285	68332	298.87	128.01	61.59
0.95	122121.6	118772	125625	0.03	122824.4	119506	126353	13.68	13.72	122968.6	119645	126533	292.13	130.89	130.92

especially for instances with a ϕ value of 0.8 and 0.85. For example, NXCC was able to reduce the average run time of this technique from 2.79 to 0.11 seconds, which represents a 25x computational speedup.

Even though the LP relaxation technique achieves the best bounds, its run time is considerably larger than the other two methods, which may make this techniques somewhat less useful in real world applications. However, incorporating the B&C method was able to reduce the run time significantly, specifically by more than half in all tested instances. Moreover, NXCC was able to accelerate the LP relaxation technique with B&C rather remarkably for instances with a ϕ value of 0.8,0.85, and 0.9. For example, it was able to reduce the average run time of this technique from 135.5 to 6.71 seconds, which represents a 20x computational speedup. It is worth mentioning that Brancotte et al. [16] introduced a formulation that could be adapted to Kemeny aggregation with non-strict rankings [18]. This means that LP relaxation bounds could be obtained from this alternative formulation as well. We conducted a separate set of experiments to evaluate the quality of these bounds and found that the LP relaxation of Brancotte's formulation provided the same bound as that of the GKPB formulation for nearly 95% of instances, and it provided a slightly better bound in most (but not all) of the remaining instances. However, since the run times of this alternative formulation were 2.6 to 4.4 times slower than GKBP, we elected not to report these results for succinctness.

It is important to remark that the maximum NXCC partitioning time over all tested instances was only 0.09 seconds, which makes this computationally inexpensive operation worthwhile to lower bounding techniques for Kemeny aggregation.

All in all, the pairwise comparison method is suitable when a very fast lower bounding technique is required. When high quality bounds are desired, the cycle-based method combined with NXCC partitioning is probably the best candidate, as it can produce competitive bounds in up to a handful of seconds. Finally, LP relaxation combined with B&C and NXCC method produces the tightest bounds in considerably more time.

VII. CONCLUSION AND FUTURE RESEARCH

This paper explores lower bounding techniques for Kemeny aggregation problem with non-strict rankings. It generalizes various existing methods for strict rankings, namely, those based on pairwise comparisons, cycles, and LP relaxations, to the case of non-strict rankings. Additionally, it utilizes partitioning using variations of the seminal Condorcet criterion and Branch & Cut (B&C) methods to accelerate the lower bounding process. The experimental results demonstrate the LP relaxation provides the tightest bounds, but it is substantially more computationally demanding than the other techniques. Deploying partitioning and B&C can drastically reduce the run time of this technique. Moreover, the cycle-based method produces high quality bounds in a reasonable time, and its run times can also be further reduced by partitioning.

Finally, future research can assess and enhance the performance of approximation algorithms [42] and exact methods [17] using the generalized lower bounding techniques developed herein. Furthermore, there might be other ways to generalize the lower bounds for Kemeny aggregation with strict rankings to the case of non-strict rankings.

Acknowledgment

The authors gratefully acknowledge funding support from the National Science Foundation (Award 1850355). This research was supported in part through high performance computing resources provided by Arizona State University.

References

[1] A. Kouadria, O. Nouali, and M. Y. H. Al-Shamri, "A multi-criteria collaborative filtering recommender system using learning-to-rank and rank aggregation," *Arabian Journal for Science and Engineering*, vol. 45, no. 4, pp. 2835–2845, 2020.

[2] S. E. Oliveira, V. Diniz, A. Lacerda, L. Merschmanm, and G. L. Pappa, "Is rank aggregation effective in recommender systems? an experimental analysis," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 11, no. 2, pp. 1–26, 2020.

[3] S. Dahiya, S. Handa, and N. Singh, "A rank aggregation algorithm for ensemble of multiple feature selection techniques in credit risk evaluation," *International Journal of Advanced Research in Artificial Intelligence*, vol. 5, no. 9, pp. 1–8, 2016.

[4] C. Sarkar, S. Cooley, and J. Srivastava, "Robust feature selection technique using rank aggregation," *Applied Artificial Intelligence*, vol. 28, no. 3, pp. 243–257, 2014.

[5] S. Cohen-Boulakia, A. Denise, and S. Hamel, "Using medians to generate consensus rankings for biological data," in *International Conference on Scientific and Statistical Database Management*. Springer, 2011, pp. 73–90.

[6] D. Marbach, J. C. Costello, R. Küffner, N. M. Vega, R. J. Prill, D. M. Camacho, K. R. Allison, M. Kellis, J. J. Collins, and G. Stolovitzky, "Wisdom of crowds for robust gene network inference," *Nature methods*, vol. 9, no. 8, pp. 796–804, 2012.

[7] J. d. Borda, "Mémoire sur les élections au scrutin," *Histoire de l'Academie Royale des Sciences pour 1781 (Paris, 1784)*, 1784.

[8] A. H. Copeland, "A reasonable social welfare function," mimeo, 1951. University of Michigan, Tech. Rep., 1951.

[9] J. G. Kemeny and L. Snell, "Preference ranking: an axiomatic approach," *Mathematical models in the social sciences*, pp. 9–23, 1962.

[10] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar, "Rank aggregation methods for the web," in *Proceedings of the 10th international conference on World Wide Web*, 2001, pp. 613–622.

[11] F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia, *Handbook of computational social choice*. Cambridge University Press, 2016.

[12] H. P. Young and A. Levenglick, "A consistent extension of condorcet's election principle," *SIAM Journal on applied Mathematics*, vol. 35, no. 2, pp. 285–300, 1978.

[13] 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*, vol. 6, no. 2, pp. 157–165, 1989.

[14] V. Conitzer, A. Davenport, and J. Kalagnanam, "Improved bounds for computing kemeny rankings," in *AAAI*, vol. 6, 2006, pp. 620–626.

[15] W. D. Cook, "Distance-based and ad hoc consensus models in ordinal preference ranking," *European Journal of operational research*, vol. 172, no. 2, pp. 369–385, 2006.

[16] B. Brancotte, B. Yang, G. Blin, S. Cohen-Boulakia, A. Denise, and S. Hamel, "Rank aggregation with ties: Experiments and analysis," *Proceedings of the VLDB Endowment (PVLDB)*, vol. 8, no. 11, pp. 1202–1213, 2015.

[17] E. J. Emond and D. W. Mason, "A new rank correlation coefficient with application to the consensus ranking problem," *Journal of Multi-Criteria Decision Analysis*, vol. 11, no. 1, pp. 17–28, 2002.

[18] Y. Yoo and A. R. Escobedo, "A new binary programming formulation and social choice property for Kemeny rank aggregation," Forthcoming in *Decision Analysis* (optimization-online preprint 7958), 2021.

[19] J. A. Aledo, J. A. Gámez, and D. Molina, "Approaching the rank aggregation problem by local search-based metaheuristics," *Journal of Computational and Applied Mathematics*, vol. 354, pp. 445–456, 2019.

[20] S. Amodio, A. D'Ambrosio, and R. Siciliano, "Accurate algorithms for identifying the median ranking when dealing with weak and partial rankings under the kemeny axiomatic approach," *European Journal of Operational Research*, vol. 249, no. 2, pp. 667–676, 2016.

[21] J. Ding, D. Han, and Y. Yang, "Iterative ranking aggregation using quality improvement of subgroup ranking," *European Journal of Operational Research*, vol. 268, no. 2, pp. 596–612, 2018.

[22] N. Ailon, M. Charikar, and A. Newman, "Aggregating inconsistent information: ranking and clustering," *Journal of the ACM (JACM)*, vol. 55, no. 5, pp. 1–27, 2008.

[23] N. Ailon, "Aggregation of partial rankings, p-ratings and top-m lists," *Algorithmica*, vol. 57, no. 2, pp. 284–300, 2010.

[24] A. Ali and M. Meilă, "Experiments with kemeny ranking: What works when?" *Mathematical Social Sciences*, vol. 64, no. 1, pp. 28–40, 2012.

[25] M. Truchon, *An extension of the Condorcet criterion and Kemeny orders*. Citeseer, 1998.

[26] N. Betzler, R. Bredereck, and R. Niedermeier, "Theoretical and empirical evaluation of data reduction for exact kemeny rank aggregation," *Autonomous Agents and Multi-Agent Systems*, vol. 28, no. 5, pp. 721–748, 2014.

[27] D. Coppersmith, L. K. Fleischer, and A. Rurda, "Ordering by weighted number of wins gives a good ranking for weighted tournaments," *ACM Transactions on Algorithms (TALG)*, vol. 6, no. 3, pp. 1–13, 2010.

[28] E. Moreno-Centeno and A. R. Escobedo, "Axiomatic aggregation of incomplete rankings," *IIE Transactions*, vol. 48, no. 6, pp. 475–488, 2016.

[29] A. Davenport and J. Kalagnanam, "A computational study of the kemeny rule for preference aggregation," in *AAAI*, vol. 4, 2004, pp. 697–702.

[30] C. Kenyon-Mathieu and W. Schudy, "How to rank with few errors," in *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, 2007, pp. 95–103.

[31] D. Younger, "Minimum feedback arc sets for a directed graph," *IEEE Transactions on Circuit Theory*, vol. 10, no. 2, pp. 238–245, 1963.

[32] R. Milosz and S. Hamel, "Exploring the median of permutations problem," *Journal of Discrete Algorithms*, vol. 52, pp. 92–111, 2018.

[33] S. I. Gass, "Tournaments, transitivity and pairwise comparison matrices," *Journal of the Operational Research Society*, vol. 49, no. 6, pp. 616–624, 1998.

[34] G. L. Nemhauser and G. Sigismondi, "A strong cutting plane/branch-and-bound algorithm for node packing," *Journal of the Operational Research Society*, vol. 43, no. 5, pp. 443–457, 1992.

[35] M. J. A. Marquis de Condorcet, *Essai sur l'application de l'analyse a la probabilite des decisions: rendues a la pluralite de voix*. De l'Imprimerie royale, 1785.

[36] H. P. Young, "Condorcet's theory of voting," *American Political science review*, vol. 82, no. 4, pp. 1231–1244, 1988.

[37] K. E. Pedings, A. N. Langville, and Y. Yamamoto, "A minimum violations ranking method," *Optimization and engineering*, vol. 13, no. 2, pp. 349–370, 2012.

[38] C. L. Mallows, "Non-null ranking models. i," *Biometrika*, vol. 44, no. 1/2, pp. 114–130, 1957.

[39] J. I. Marden, *Analyzing and Modeling Rank Data*. CRC Press, 2014.

[40] J.-P. Doignon, A. Pekeč, and M. Regenwetter, "The repeated insertion model for rankings: Missing link between two subset choice models," *Psychometrika*, vol. 69, no. 1, pp. 33–54, 2004.

[41] Y. Yoo, A. R. Escobedo, and J. K. Skolfield, "A new correlation coefficient for comparing and aggregating non-strict and incomplete rankings," *European Journal of Operational Research*, 2020.

[42] S. Akbari and A. R. Escobedo, "Approximate condorcet partitioning: Solving very large rank aggregation problems at scale," *Under review*, 2021.