Weighted Rank Aggregation via Relaxed Integer Programming

Fardad Raisali, **Farzad Farnoud (Hassanzadeh)**, **Olgica Milenkovic**
Department of Electrical and Computer Engineering, University of Illinois, Urbana Champaign, U.S.A.
{raisali2, hassanz1, milenkov}@illinois.edu

***Abstract*—We propose a new family of algorithms for bounding/approximating the optimal solution of rank aggregation problems based on weighted Kendall distances. The algorithms represent linear programming relaxations of integer programs that involve variables reflecting partial orders of *three or more candidates*. Our simulation results indicate that the linear programs give near-optimal performance for a number of important voting parameters, and outperform methods based on PageRank and Weighted Bipartite Matching.**

I. Introduction

The problem of rank aggregation may be simply stated as follows: a set of voters or agents is presented with a list of candidates that have to be ranked according to some criteria. The aggregate ranking is chosen to best reflect the ordering provided by the voters. Due to the fact that large volume datasets in social science, search engines, and biology are ordinal data, frequently obtained from multiple sources and using different ranking functions, rank aggregation has found many applications in web metasearch engines, social sciences, spam control and other applications [8], [9].

One of the best known methods for rank aggregation is distance based aggregation, where the problem is cast as the computation of the median of a set of full rankings (permutations). The distance measure used for computing the median is the Kendall distance, which has also found many applications outside of social choice theory and computer science – for example, in rank modulation coding for flash memories [10]. The Kendall distance counts the number of pairwise disagreements between two permutations ([1], [2]), and can be computed efficiently. On the other hand, computing the aggregate ranking under the Kendall distance is known to be NP hard [7]. To overcome this computational bottleneck, a number of algorithms for approximate aggregation were put forward, including PageRank (PR), Weighted Bipartite Graph Matching (WBGM), and relaxed Integer Programming (IP) (in particular, linear programing (LP) methods) [8], [9].

PR methods for rank aggregation mimic the principles used for ranking webpages by Google, and they reduce to computing equilibrium probabilities of Markov chains. WBGM algorithms utilize the fact that the Kendall distance may be approximated up to a multiplicative constant by the ℓ_1 norm of permutations. The close connection between transitive tournaments and rankings was the basis for developing IP aggregation methods [4].

It is well known that the Kendall distance is not suitable for many practical applications in which human subjects are involved, since the Kendall distance does not account for the fact that one inevitably pays more attention to the top of a list then to the remainder of the list. To overcome this problem, in our recent work we introduced the notion of a *weighted Kendall distance*, where higher weights are assigned to adjacent swaps at the top of a list. This ensures that in an aggregate, strong showings of candidates are emphasized compared to their weaker showings. In a companion paper [11], we presented extensions of the PR and WBGM methods for weighted Kendall distances. In what follows, we present a novel combinatorial optimization framework for computing the weighted Kendall aggregate with near-optimal performance. The algorithm is based on a new representation of permutations using partial orderings of three or more candidates as constraints. The method is of especially simple form when the weights are monotonically decreasing functions, and we therefore focus our attention to this case. Decreasing weights are suitable for capturing the importance of the top of a list, as they ensure that changes at the top are costlier than changes at the bottom.

The paper is organized as follow. In Section II, we introduce the Kendall and weighted Kendall distance, an IP problem formulation of Kendall distance based aggregation, and its corresponding LP relaxation. In Section III, we derive a closed form expression for linearly decreasing weighted Kendall distances, describe a corresponding IP aggregation method, and its LP relaxation. We also describe how this approach may be viewed as a special scoring procedure on rankings. Section IV contains extensions of the aforementioned results to the case of polynomially decreasing weight functions, while simulation results are given in Section V.

II. Notation and Preliminaries

We consider the problem of rank aggregation involving n candidates and m voters. For simplicity, the set of candidates is chosen as $\{1, \ldots, n\}$, and denoted by $[n]$. A vote is a ranking of the candidates with no ties, and hence a permutation in $\mathbb{S}_n$, the symmetric group of order $n!$. We write each permutation $\sigma \in \mathbb{S}_n$ as $\sigma(1) \cdots \sigma(n)$, where $\sigma(i)$ represents the candidate with rank i. Note that $\sigma^{-1}(i)$ is the rank of candidate i, where σ^{-1} denotes the inverse of σ.

Suppose that the voters are numbered from 1 to m. Voters are allowed to cast the same vote, and the multiset of the voters' permutations (rankings) is denoted by Σ.

In *distance-based* rank aggregation, the goal is to find a ranking, called the *aggregate ranking*, that is as "close" as possible to all the votes simultaneously. Closeness is measured via a chosen distance function over $\mathbb{S}_n$. For a given distance d, the aggregate ranking π is formally evaluated according to

$$\pi^* = \arg\min_{\pi \in \mathbb{S}_n} \sum_{\sigma \in \Sigma} d(\pi, \sigma). \tag{1}$$

The most commonly used distance for the purpose of rank aggregation is the Kendall distance, although other distances, such as the Cayley distance, Spearman's footrule, and Spearman's rank correlation have found relevant applications [6]. The Kendall distance between two permutations π and σ, denoted by $\mathrm{d}_K(\pi, \sigma)$, is the number of disagreements between π and σ, i.e., the number of ordered pairs (i, j) such that π ranks i higher than j, and σ ranks j higher than i. Formally, the distance may be defined as

$$\mathrm{d}_K(\pi, \sigma) = \left|\{(i,j) : \pi^{-1}(i) < \pi^{-1}(j), \sigma^{-1}(j) < \sigma^{-1}(i)\}\right|.$$

The solution of (2) for the Kendall distance is known as the *Kemeny aggregate*.

For $\sigma \in \mathbb{S}_n$, and $i, j \in [n]$, let

$$\sigma_{ij} = \begin{cases} 1, & \text{if } \sigma^{-1}(i) < \sigma^{-1}(j), \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Let P be the set of points $x = (x_{ij})$ satisfying

$$x_{ij} + x_{ji} = 1, \qquad \text{for distinct } i, j \in [n], \tag{3}$$
$$x_{ij} + x_{jk} + x_{ki} \leq 2, \qquad \text{for distinct } i, j, k \in [n], \tag{4}$$
$$x_{ij} \in \{0, 1\}, \qquad \text{for distinct } i, j \in [n], \tag{5}$$
$$x_{ii} = 0, \qquad \text{for } i \in [n]. \tag{6}$$

Note that there is a one-to-one correspondence between points $x \in P$ and permutations $\pi \in \mathbb{S}_n$, since $\pi^{-1}(i) < \pi^{-1}(j)$ if and only if $x_{ij} = 1$.

Using (2) and the definition of the Kendall distance, for each $x \in P$, one can write

$$\sum_{\sigma \in \Sigma} \mathrm{d}_K(x, \sigma) = \sum_{\sigma \in \Sigma} \sum_{i,j} x_{ij}\sigma_{ji} = \sum_{i,j} c_{ij} x_{ij}, \tag{7}$$

where $c_{ij} = \sum\limits_{\sigma \in \Sigma} \sigma_{ji}$.

From (7) and the fact that the constraints (3)-(5) define a permutation, we find that a Kemeny aggregate is a solution of the integer programing (IP) problem

$$\min_x \sum_{\sigma \in \Sigma} \sum_{i,j} c_{ij} x_{ij}$$
$$\text{subject to } x_{ij} \in P.$$

This formulation was independently proposed in [9], while relaxations of the IP method were shown to provide good approximations to the exact solution in [12].

In what follows, we describe how to generalize this simple idea for a broad class of *weighted* Kendall distance measures. Weighted Kendall distances were introduced by the authors in [11], and may be defined as follows. An adjacent transposition in a permutation is a swap of two elements ranked consecutively. Endow the set of adjacent transpositions A with a weight function $\rho : A \to \mathbb{R}^+$, i.e., assign to each adjacent transposition $(i\; i+1)$ a non-negative weight ρ_i.

The weighted Kendall distance under ρ, applied to two permutations π and σ, equals the smallest cost of any sequence of adjacent transpositions needed to transform π into σ. For example, let $\rho_1 = 2$ and $\rho_2 = 1$. The weighted Kendall distance between 132 and 213 equals $\rho_2 + \rho_1 = 3$, since one may first swap candidates 2 and 3 with weight ρ_2, and then swap candidates 2 and 1 with weight ρ_1.

In many applications, the top of a ranking is more important than the bottom, and thus it is reasonable to require that changes to the top of a ranking induce a larger distance than similar changes applied to the bottom of a ranking. Unfortunately, the classical Kendall distance does not take into account positional significance of candidates in a ranking, as any adjacent transposition contributes one point to the total distance. Weighted distances can overcome this problem, since they do not require uniform weights for adjacent swaps.

A. Problem Reformulation

In what follows, we describe an alternative formulation for P that will be useful in our subsequent analysis.

Let $\mathcal{T}_{a,b,c} = \{(abc), (acb), (bac), (bca), (cba), (cab)\}$. In addition, let Q be the set of points (x, w), with $x = (x_{ij})$, $i, j \in [n]$, and $w = (w_{ijk})$, with $i, j, k \in [n]$, satisfying

$$\sum\nolimits_{(rst) \in \mathcal{T}_{i,j,k}} w_{rst} = 1, \qquad \text{for distinct } i, j, k \in [n], \tag{8}$$
$$w_{ijk} + w_{ikj} + w_{kij} = x_{ij}, \qquad \text{for distinct } i, j, k \in [n], \tag{9}$$
$$x_{ij},\, w_{ijk} \in \{0, 1\}, \qquad \text{for distinct } i, j, k \in [n], \tag{10}$$
$$w_{ijk} = 0, \qquad \text{for } i, j, k \text{ not distinct.} \tag{11}$$

Note that there is a one-to-one correspondence between points $(x, w) \in Q$ and permutations $\pi \in \mathbb{S}_n$, where $x_{ij} = 1$ if and only if $\pi^{-1}(i) < \pi^{-1}(j)$, and $w_{ijk} = 1$ if and only if $\pi^{-1}(i) < \pi^{-1}(j) < \pi^{-1}(k)$.

Define $\bar{Q}$ similarly to Q, by replacing the integrality condition (10) with $0 \leq w_{ijk} \leq 1$. In other words, let $\bar{Q}$ be the convex hull of Q. Clearly, $\bar{Q}$ is a polytope. Also, define $\bar{P}$ by replacing (5) with $0 \leq x_{ij} \leq 1$ in the definition of P. Finally, let $Q_p = \{x : (x, w) \in Q\}$ and $\bar{Q}_p = \{x : (x, w) \in \bar{Q}\}$.

Theorem 1: The sets P and Q_p are identical.

Proof: We first show that $x \in Q_p$ implies $x \in P$. For $x \in Q_p$ and distinct $i, j, k \in [n]$, one has

$$x_{ij} + x_{ji} = \sum\nolimits_{(rst) \in \mathcal{T}_{i,j,k}} w_{rst} = 1,$$

where the first equality follows from (9) and the second equality follows from (8). This proves (3).

To prove (4), for distinct $i, j, k \in [n]$, one may write

$$\begin{aligned} x_{ij} + x_{jk} + x_{ki} &= w_{ijk} + w_{ikj} + w_{kij} + w_{jki} + w_{jik} \\ &\quad + w_{ijk} + w_{kij} + w_{kji} + w_{jki} \\ &= 1 + w_{kji} + w_{kij} + w_{jki} \leq 2, \end{aligned}$$

where the first equality follows from (9), while the other two equalities follow from (8).

From (8) and (9), one has $x_{ij} \leq 1$, and from (9) and (10), it follows that x_{ij} is a non-negative integer. Hence, x_{ij} is either 0 or 1, proving (5). To complete the proof of the claim that $Q_p \subset P$, observe that (6) follows from (9) and (11).

Suppose next that $x \in P$. For $i, j, k \in [n]$, let $w_{ijk} = x_{ij}x_{jk}$. We show that $x \in Q_p$ by proving that $(x, w) \in Q$. It is clear that (10) is satisfied.

When $i = j$ or $j = k$, the proof of (11) follows from (6). If $i = k \neq j$, then (11) follows from (3).

To see that (9) holds, note that, for distinct $i, j, k \in [n]$,

$$x_{ij} = x_{ij}x_{jk} + x_{ik}x_{kj} + x_{ki}x_{ij} = w_{ijk} + w_{ikj} + w_{kij}.$$

The first equality can be verified by considering all possible choices for (x_{ij}, x_{jk}, x_{ki}), i.e. by observing that

$$\begin{aligned}(x_{ij}, x_{jk}, x_{ki}) \in \big\{&(0,0,1),(0,1,0),\\ &(1,0,0),(0,1,1),(1,1,0),(1,0,1)\big\},\end{aligned}$$

since $(x_{ij}, x_{jk}, x_{ki}) = (0,0,0)$ and $(x_{ij}, x_{jk}, x_{ki}) = (1,1,1)$ are excluded by (4). As a result, (8) follows from (3) and (9). ∎

Theorem 2: The sets $\bar{P}$ and $\bar{Q}_p$ are identical.

Proof: (Sketch) Each triple (r, s, t) appears in the definition of $\bar{Q}_p$ as part of the following constraints:

$$\begin{aligned}\textstyle\sum_{(ijl)\in\mathcal{T}_{r,s,t}} w_{ijl} = 1,&\\ w_{ijl} + w_{ilj} + w_{lij} = x_{ij}, \quad \forall (ijl) \in \mathcal{T}_{r,s,t}.&\end{aligned} \tag{12}$$

Similarly, each triple (r, s, t) appears in the definition of $\bar{P}_p$ as part of the following constraints:

$$\begin{aligned}x_{rs} + x_{st} + x_{tr} \leq 2,&\\ x_{sr} + x_{ts} + x_{rt} \leq 2,&\\ x_{ij} + x_{ji} = 1 \quad \forall (ijl) \in \mathcal{T}_{r,s,t}.&\end{aligned} \tag{13}$$

Consider the tuples $(x_{rs}, x_{st}, x_{tr}, x_{sr}, x_{ts}, x_{rt})$ as restricted by 12 and 13, and denote them by $\bar{P}^{rst}$ and $\bar{Q}_p^{rst}$, respectively.

We first show that $\hat{x} \in \bar{Q}_p^{rst}$ implies $\hat{x} \in \bar{P}^{rst}$. Note that $\bar{Q}_p^{rst}$ is the convex hull of the points

$$(1,1,0,0,0,1), (0,1,1,1,0,0), (1,0,1,0,1,0),$$

$$(0,0,1,1,1,0), (1,0,0,0,1,1), (0,1,0,1,0,1).$$

It is easy to check that these points belong to $\bar{P}^{rst}$ as well, which completes the claim.

Next, we show that $\hat{x} \in \bar{P}^{rst}$ implies $\hat{x} \in \bar{Q}_p^{rst}$. Assume that there exists a $\hat{x} \in \bar{P}^{rst}$ such that $\hat{x} \notin \bar{Q}_p^{rst}$. Since $\hat{x} \notin \bar{Q}_p^{rst}$, there exists a facet of $\bar{Q}_p^{rst}$ which serves as a separating hyperplane between $\hat{x}$ and the interior of the polytope. Moreover, this facet is also a separating hyperplane for at least one vertex of the unit cube which does not belong to the convex hull [3]. Note that the vertices of the unit cube that do not belong to the convex hull are

$$(1,1,1,*,*,*), (*,*,*,1,1,1), (1,*,*,1,*,*), (0,*,*,0,*,*),$$

$$(*,1,*,*,1,*), (*,0,*,*,0,*), (*,*,1,*,*,1), (*,*,0,*,*,0);$$

the symbol "$*$" stands for either 1 or 0.

The facet $x_{rs} + x_{st} + x_{tr} = 2$ is a separating hyperplane for $(1,1,1,*,*,*)$. The three vertices of the facet are $(1,1,0,0,0,1)$, $(0,1,1,1,0,0)$ and $(1,0,1,0,1,0)$, and for all points in the polytope not incident with the facet we have $x_{rs} + x_{st} + x_{tr} < 2$. Since $\hat{x}$ is assumed not to belong to $\bar{Q}_p^{rst}$, it must hold that $\hat{x}_{rs} + \hat{x}_{st} + \hat{x}_{tr} > 2$. But this contradicts the assumption that $\hat{x} \in \bar{P}^{rst}$.

The proof follows by considering all other vertices of the unit cube. ∎

III. Weighted Rank Aggregation

A. Linear Weighted Distances

While an efficient algorithm for computing the weighted Kendall distance with an arbitrary weight function ρ is not known, a polynomial-time algorithm exists if the weight function is decreasing, i.e., if $\rho_i \geq \rho_{i+1}$.

Consider the following linear weight function:

$$\rho_i = 1 + \frac{\epsilon}{n-2}(n-1-i), \tag{14}$$

where $\epsilon \geq 0$. This function assigns weight $1+\epsilon$ to a swap involving the first and the second candidate, and weight 1 to a swap involving the last and the next to last candidate. The weights decrease linearly between these two points. Note that with this choice, swapping candidates at the top induces a larger distance between permutations. We subsequently make use of the following weight functions as well,

$$\rho_i = 1 + \frac{\epsilon}{n-2}(n-1-i)^k, \tag{15}$$

where k is a positive integer, and $\epsilon > 0$.

Let $I(\pi, \sigma)$ denote the set of ordered pairs (a, b) for which $\pi^{-1}(a) < \pi^{-1}(b)$ and $\sigma^{-1}(b) < \sigma^{-1}(a)$.

Lemma 1: For permutations $\pi, \sigma \in \mathbb{S}_n$, and the weight function ρ of (14), we have

$$\mathrm{d}_\rho(\pi, \sigma) = \sum_{i,j} \pi_{ij}\sigma_{ji}\Big(1 + \frac{\epsilon}{n-2}\sum_k \pi_{ik}\sigma_{jk}\Big). \tag{16}$$

Proof: (Sketch) It was shown in [11] that the minimum weight sequence of adjacent transpositions that converts π to σ is obtained as follows: for $\ell = 1, \ldots, n$, find $\sigma(\ell)$ in π and move it to position ℓ in π using adjacent transpositions. It then follows that the transposition that swaps $(i, j) \in I(\pi, \sigma)$ has weight ρ_s, where

$$s = \pi^{-1}(i) + \left|\left\{k : \sigma^{-1}(k) < \sigma^{-1}(j), \pi^{-1}(i) < \pi^{-1}(k)\right\}\right|.$$

It is not hard to show that s can also be written as

$$s = n - 1 - \left|\left\{k : \pi^{-1}(i) < \pi^{-1}(k), \sigma^{-1}(j) < \sigma^{-1}(k)\right\}\right|.$$

Using (2), we have $s = n-1-\sum_k \pi_{ik}\sigma_{jk}$. The lemma follows from (14). ∎

The objective function of the rank aggregation problem (1), with weights given by (14), equals

$$\begin{aligned}\sum_{\sigma\in\Sigma} \mathrm{d}_\rho(x, \sigma) &= \sum_{\sigma\in\Sigma}\sum_{i,j} x_{ij}\sigma_{ji}\Big(1 + \frac{\epsilon}{n-2}\sum_k x_{ik}\sigma_{jk}\Big)\\ &= \sum_{i,j} x_{ij}\sum_{\sigma\in\Sigma}\sigma_{ji} + \frac{\epsilon}{n-2}\sum_{i,j,k} x_{ij}x_{ik}\sum_{\sigma\in\Sigma}\sigma_{ji}\sigma_{jk}.\end{aligned} \tag{17}$$

Let d_{ijk} denote the number of voters who prefer i to j, and j to k. Note that $\sum_{\sigma\in\Sigma}\sigma_{ji}\sigma_{jk}=d_{jik}+d_{jki}$. Hence, for $x\in P$,

$$\sum_{\sigma\in\Sigma}\mathrm{d}_\rho(x,\sigma)=\sum_{i,j}c_{ij}x_{ij}+\frac{\epsilon}{n-2}\sum_{i,j,k}(d_{jik}+d_{jki})\,x_{ij}x_{ik}.$$

The objective function consequently reduces to

$$\min_{x\in P}\sum_{i,j}c_{ij}x_{ij}+\frac{\epsilon}{n-2}\sum_{i,j,k}(d_{jik}+d_{jki})\,x_{ij}x_{ik}. \tag{18}$$

Theorem 1 implies that $x\in P$ if and only if $x\in Q_p$. Hence, one can replace $x\in P$ in (18) with $(x,w)\in Q$. For every $(x,w)\in Q$ and $i,j,k\in[n]$, it is straightforward to see that $x_{ij}x_{ik}=w_{ijk}+w_{ikj}$. Hence, we may rewrite (18) as

$$\min_{(x,w)\in Q}\sum_{i,j}c_{ij}x_{ij}+\frac{\epsilon}{n-2}\sum_{i,j,k}(d_{jik}+d_{jki})\,(w_{ijk}+w_{ikj})\,. \tag{19}$$

Since $c_{ij}=d_{jik}+d_{jki}+d_{kji}$, and

$$x_{ij}=\frac{1}{n-2}\sum_k(w_{ijk}+w_{ikj}+w_{kij})\,,$$

it is apparent that (19) is equivalent to

$$\min_{w\in W}\frac{1}{n-2}\sum_{i,j,k}\alpha_{ijk}w_{ijk}, \tag{20}$$

where $W=\{w:(x,w)\in Q\}$ and

$$\begin{aligned}\alpha_{ijk}=d_{ikj}&+(1+\epsilon)d_{jik}+(2+\epsilon)d_{kij}\\&+(2+\epsilon)d_{jki}+(3+\epsilon)d_{kji}.\end{aligned}$$

The coefficients on the right side of the above equation have an interesting interpretation. For each permutation (rst) of $\{i,j,k\}$, the coefficient of d_{rst} equals the weighted Kendall distance between the permutations (rst) and (ijk), based on the weight function (14) and for $n=3$. In other words,

$$\alpha_{ijk}=\sum_{(rst)\in\mathcal{T}_{i,j,k}}\mathrm{d}_\rho(rst,ijk)\,d_{rst},$$

which for $\epsilon=1$ reduces to

$$\alpha_{ijk}=d_{ikj}+2d_{jik}+3d_{kij}+3d_{jki}+4d_{kji}.$$

B. The Dual Problem

The dual of the problem (20) can be written as

$$\begin{aligned}&\max_\lambda\sum_{i<j<k}\lambda_{\{i,j,k\}}\\&\quad\text{s.t. for all distinct } i,j,k\in[n]:\\&\qquad\lambda_{\{i,j,k\}}+\nu_{ijk}+\nu_{ikj}+\nu_{jki}\\&\qquad-\nu_{ijh_{ij}(k)}-\nu_{ikh_{ik}(j)}-\nu_{jkh_{jk}(i)}\le\alpha_{ijk},\end{aligned}$$

The brackets in the subscript of λ indicate that $\lambda_{\{i,j,k\}}=\lambda_{\{i,k,j\}}=\cdots$, i.e., that the order of $i,j,$ and k does not matter. Here, $h_{ij}(k)$ is the element that (circularly) precedes k in the vector $(1,\cdots,i-1,i+1,\cdots,j-1,j+1,\cdots,n)$. For example, $h_{25}(4)=3$ and $h_{12}(3)=n$.

There does not seem to be a clear interpretation of the dual problem. However, if we let the ν variables equal to zero, we obtain the following problem:

$$\max\sum_{i<j<k}\lambda_{\{i,j,k\}} \tag{21}$$

$$\text{s.t. }\lambda_{\{i,j,k\}}\le\min\{\alpha_{rst}:(rst)\in\mathcal{T}_{i,j,k}\},\quad\forall\ i<j<k.$$

The optimal value of the latter problem has a clear interpretation as a lower bound: for each set of distinct values $\{i,j,k\}$ at least one of the w's is one, and thus at least a value of $\min\{\alpha_{ijk},\alpha_{ikj},\alpha_{kij},\alpha_{jik},\alpha_{jki},\alpha_{kji}\}$ is contributed to the total sum.

IV. Quadratic Weight Functions

In Section III, we derived a linear programming relaxation of the rank aggregation problem with the linear weight function

$$\rho_i=1+\frac{\epsilon}{n-2}(n-1-i).$$

A similar approach can be used for weight functions of the more general form of (15), with k a positive integer. For simplicity, we illustrate the general problem on the quadratic weight function

$$\rho_i=1+\frac{\epsilon}{n-2}(n-1-i)^2. \tag{22}$$

For the quadratic weight function ρ, the distance between rankings π and σ is

$$\mathrm{d}_\rho(\pi,\sigma)=\sum_{i,j}\pi_{ij}\sigma_{ji}\Big(1+\frac{\epsilon}{n-2}\big(\sum_k\pi_{ik}\sigma_{jk}\big)^2\Big). \tag{23}$$

Hence, for $x\in P$,

$$\begin{aligned}\sum_{\sigma\in\Sigma}\mathrm{d}_\rho(x,\sigma)&=\sum_{\sigma\in\Sigma}\sum_{i,j}x_{ij}\sigma_{ji}\Big(1+\frac{\epsilon}{n-2}\big(\sum_k x_{ik}\sigma_{jk}\big)^2\Big)\\&=\sum_{\sigma\in\Sigma}\sum_{i,j}x_{ij}\sigma_{ji}+\sum_{\sigma\in\Sigma}\sum_{i,j,k}\frac{\epsilon}{n-2}x_{ik}x_{ij}\sigma_{jk}\sigma_{ji}\\&\quad+\sum_{\sigma\in\Sigma}\sum_{i,j,k}\sum_{l\neq k}\frac{\epsilon}{n-2}x_{ik}x_{il}x_{ij}\sigma_{jk}\sigma_{jl}\sigma_{ji}.\end{aligned}$$

Let R be the set of points (x,w), with $x=(x_{ij})$ and $w=(w_{ijkl})$, satisfying

$$\begin{aligned}\textstyle\sum_{(rstu)\in\mathcal{T}_{i,j,k,l}}w_{rstu}&=1, &&\text{for distinct } i,j,k,l\in[n],\\\textstyle\sum_{(rstu)\in\mathcal{T}^{i>j}_{i,j,k,l}}w_{rstu}&=x_{ij}, &&\text{for distinct } i,j,k\in[n],\\w_{ijkl}&\in\{0,1\}, &&\text{for distinct } i,j,k,l\in[n],\\w_{ijkl}&=0, &&\text{for } i,j,k,l \text{ not distinct},\end{aligned}$$

where $\mathcal{T}_{i,j,k,l}$ denotes the set of permutations of $\{i,j,k,l\}$ and $\mathcal{T}^{i>j}_{i,j,k,l}$ denotes the set of permutations of $\{i,j,k,l\}$ in which i appears before j. Note that there is a one-to-one correspondence between points $(x,w)\in R$ and permutations $\pi\in\mathbb{S}_n$, where $x_{ij}=1$ if and only if $\pi^{-1}(i)<\pi^{-1}(j)$ and $w_{ijkl}=1$ if and only if $\pi^{-1}(i)<\pi^{-1}(j)<\pi^{-1}(k)<\pi^{-1}(l)$.

Similar to Theorem 1, one can show that $P=\{x:(x,w)\in R\}$. Furthermore, it is straightforward to show that

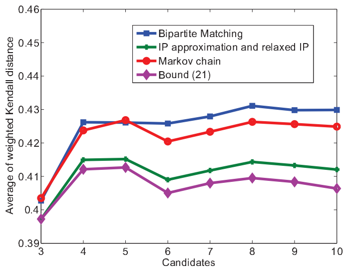

Fig. 1: m=10.

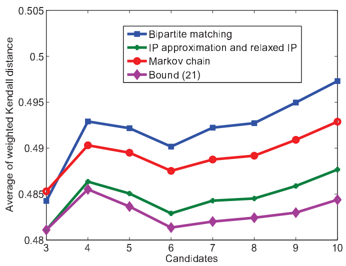

Fig. 2: m=50.

$x_{ik}x_{ij}$ and $x_{ik}x_{ij}x_{il}$ are linear in $w_{rstu}, r, s, t, u \in [n]$. Let e_{ijkl} be the number of permutations $\sigma \in \Sigma$ with $\sigma^{-1}(i) < \sigma^{-1}(j) < \sigma^{-1}(k) < \sigma^{-1}(l)$. The rank aggregation problem with quadratic weight function is equivalent to

$$\arg \min_{(x,w)\in R} \sum_{i,j,k,l} \beta_{ijkl} w_{ijkl}, \tag{24}$$

where β_{ijkl}, for $i, j, k, l \in [n]$, are linear combinations of $e_{rstu}, r, s, t, u \in [n]$. Note that the objective function of (24) is linear. Furthermore, if we replace the integrality condition $w_{ijkl} \in \{0, 1\}$, for $i, j, k, l \in [n]$, with $0 \leq w_{ijkl} \leq 1$, for $i, j, k, l \in [n]$, we obtain a linear programming relaxation for the rank aggregation problem with a quadratic weight function.

V. Simulations

We evaluate the performance of the bound (21), the IP approximation (20) and relaxed IP bound (when condition (10) is replaced with $0 \leq w_{ijk} \leq 1$). Moreover, we considered the WBM and Markov chain (PR) methods, adapted for the weighted Kendall distance measures in [13]. We compared the averages of the objective function based on the weighted Kendall distance given in section III (here $\epsilon = 1$). The average value refers to

$$\frac{1}{m} \sum_{\sigma \in \Sigma} d_\rho(\hat{\pi}, \sigma),$$

where $\hat{\pi}$ represents a solution found by a particular algorithm. The minimum of the average value is attained by the optimal solution. Note that in relaxed IP and for the bound (21), the solutions do not necessarily represent permutations. In these cases, we use a lower bound on the average value of the optimal solution based on the weighted Kendall distance.

We generated different set of votes with varying number of candidates. The votes were chosen in an iid manner and are generated uniformly. The number of candidates varies from $n = 3$ to $n = 10$. For $m = 10, 50$ the results obtained by the aforementioned algorithms are depicted in Figures 1 and 2. More precisely, Figures 1 and 2 illustrate the average value of solutions obtained by IP approximation, bipartite matching, and the Markov chain method. They also illustrate lower bounds on the optimal average value obtained from (21), and from the relaxed integer programming approach. For each data point, we created 500 samples of votes.

To find the solution for the IP approximation, we used a branch and bound method. Notice that the curves for the integer programming approximation and the relaxed integer program match very well. This means that integer programming approximations are quite successful in finding the correct optimal solution based on the weighted Kendall distance. Integer programming approximations outperform bipartite matching and Markov chain techniques. The bound (21) remains below the other curves, as expected. Surprisingly, it does not exhibit large deviations from the optimal average value. This is interesting, since the bound (21) is attained with much smaller computational cost.

Acknowledgment

The authors are grateful to Angelia Nedich for many useful discussions. The work as partially supported by NSF grants CCF 0939370, CCF 0644427 and 1218764.

References

[1] J. G. Kemeny, "Mathematics without numbers," *Daedalus*, vol. 88, no. 4, pp. 577–591, 1959.
[2] M. Kendall, *Rank correlation methods*. London: Griffin, fourth ed., 1970.
[3] R. G. Jeroslow, "On defining sets of vertices of the hypercube by linear inequalities," *Discrete Mathematics*, vol. 11, pp. 119–124, 1975.
[4] G. Reinelt, "The linear ordering problem: algorithms and applications, Research and Exposition in Mathematics 8," *Berlin: Heldermann*, 1985.
[5] I. Ali, W.D. Cook, and M. Kress, "On a minimum violations ranking of a tournament,"*Manage. Sci.*, vol. 32, no. 6, pp. 660–672, June 1986.
[6] P. Diaconis, "Group representations in probability and statistics," *Lecture Notes-Monograph Series*, vol. 11, 1988.
[7] J. Bartholdi, C. Tovey, and M. Trick, "The computational difficulty of manipulating an election," *Social Choice and Welfare*, vol. 6, no. 3, pp. 227–241, 1989.
[8] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar, "Rank aggregation revisited." Manuscript, Available: www.eecs.harvard.edu/~michaelm/CS222/rank2.pdf, 2001.
[9] V. Conitzer, A. Davenport, and J. Kalagnanam, "Improved bounds for computing Kemeny rankings," in *Proc. of the 21st National Conf. on Artificial Intelligence*, (Boston, Massachusetts), 2006.
[10] A. Barg and A. Mazumdar, "Codes in permutations and error correction for rank modulation," *IEEE Trans. Information Theory*, vol. 56, pp. 3158–3165, July 2010.
[11] F. Farnoud, B. Touri, and O. Milenkovic, "Novel distance measures for vote aggregation," *arXiv preprint arXiv:1203.6371*, 2012.
[12] K. Pedings, A. Langville, and Y. Yamamoto, "A minimum violations ranking method," *Optimization and Engineering*, vol. 13, no. 2, pp. 349–370, 2012.
[13] F. Farnoud, B. Touri, and O. Milenkovic, "Nonuniform vote aggregation algorithms," in *Int. Conf. Signal Processing and Communications (SPCOM)*, Bangalore, India, July 2012.