Available online at www.sciencedirect.com
ScienceDirect
ELSEVIER
Discrete Mathematics 309 (2009) 1813–1823
DISCRETE MATHEMATICS
www.elsevier.com/locate/disc

On the complexity of crossings in permutations[☆]

Therese Biedl[a], Franz J. Brandenburg[b,*], Xiaotie Deng[c]

[a] *School of Computer Science, University of Waterloo, ON N2L3G1, Canada*
[b] *Lehrstuhl für Informatik, Universität Passau, 94030 Passau, Germany*
[c] *Department of Computer Science, City University of Hong Kong, 83 Tat Chee Avenue, Kowloon Tong, Hong Kong, China*

Received 31 January 2006; accepted 18 December 2007
Available online 12 February 2008

Abstract

We investigate crossing minimization problems for a set of permutations, where a crossing expresses a disarrangement between elements. The goal is a common permutation π^* which minimizes the number of crossings. In voting and social science theory this is known as the Kemeny optimal aggregation problem minimizing the Kendall-τ distance. This rank aggregation problem can be phrased as a one-sided two-layer crossing minimization problem for a series of bipartite graphs or for an edge coloured bipartite graph, where crossings are counted only for monochromatic edges. We contribute the max version of the crossing minimization problem, which attempts to minimize the discrimination against any permutation. As our results, we correct the construction from [C. Dwork, R. Kumar, M. Noar, D. Sivakumar, Rank aggregation methods for the Web, Proc. WWW10 (2001) 613–622] and prove the NP-hardness of the common crossing minimization problem for $k = 4$ permutations. Then we establish a $2 - 2/k$-approximation, improving the previous factor of 2. The max version is shown NP-hard for every $k \geq 4$, and there is a 2-approximation. Both approximations are optimal, if the common permutation is selected from the given ones. For two permutations crossing minimization is solved by inspecting the drawings, whereas it remains open for three permutations.

Keywords: Crossing minimization; Rank aggregation; Kendall-τ distance; NP-hardness approximations

1. Introduction

The rank aggregation problem consists of finding a consensus ranking on a set of alternatives, based on preferences of individual voters. More precisely, given a set of (possibly partial) permutations of the alternatives, decide on one permutation (called the "common permutation") of the alternatives that best captures the preferences. The roots for a mathematical investigation of the problem lie in voting theory and go back to Borda (1781) and Condorcet (1785). Rank aggregations occur in many contexts, including sports, voting, business, and most recently, the Internet. Questions such as "*What are the top i items?*" are asked frequently. One simple approach is to average the points of the

☆ The work of the first author was supported by NSERC, and done while the author was visiting Universität Passau. The work of the second and third authors was partially supported by a grant from the German Academic Exchange Service (Project D/0506978) and from the Research Grant Council of the Hong Kong Joint Research Scheme (Project No. G_HK008/04). A preliminary version has appeared in the Proceedings of GD2005.
* Corresponding author.
E-mail addresses: biedl@uwaterloo.ca (T. Biedl), brandenb@informatik.uni-passau.de (F.J. Brandenburg), csdeng@cityu.edu.hk (X. Deng).

0012-365X/$ - see front matter
doi:10.1016/j.disc.2007.12.088

judges, as done in gymnastics, figure skating or dancing. Another approach is to declare the winner to be the one with the most points, as done in Formula 1 racing and at the annual European Song Contest. Finally, there is a ranking of golf professionals by the money list, which sums the won prize money, and is a weighted aggregation of their ranking in tournaments. *"Are these schemes fair?"* Who shall be the true winner, if one candidate is three times in first place and twice in third place, and another candidate is once in first place and four times in second place? Conflicts occur if voters disagree on items and in particular, if there are cycles in the votes. Such questions are discussed in social science theory, see, e.g., [3].

We study here one version of the rank aggregation problem that can be shown to be closely related to the one-sided crossing minimization problem of two-layered bipartite graphs. Assume we want to find a common permutation such that the order of items is the same for the inputs as often as possible. Put differently, a disagreement occurs if the order of two items is different in one input permutation than in the common permutation, and we want to find a common permutation that minimizes the number of such disagreements. This problem then becomes a combination of crossing minimization problems as follows: Every input permutation is represented as a bipartite graph with the items in sorted order on one side and the common permutation on the other side. The rank aggregation problem then consists of choosing the common permutation that minimizes the total number of crossings. One-sided crossing minimization is a major component in the Sugiyama algorithm, the most popular algorithm for hierarchical graph drawing, and is one of the most intensively studied problems in graph drawing [7,14]. For general graphs the crossing minimization problem is known to be NP-hard [12]. The NP-hardness also holds for bipartite graphs where the upper layer is fixed, and the graphs are dense with $O(n^2)$ crossings [9], or alternatively, the graphs are sparse with degree at least four on the free layer [16]. The special case with degree 2 vertices on the free layer is solvable in linear time, whereas the degree 3 case is open.

In their seminal paper from the WWW10 conference, Dwork et al. [8] have used rank aggregation methods for web searching and spam reduction. A search engine is called *good* if it behaves close to the aggregate ranking of several search engines. Besides experimental results they have investigated the theoretical foundations of the rank aggregation problem. One of the main results is the NP-hardness of computing a so-called Kemeny optimal permutation of just four permutations, here called PCM-4. However, the given proof for $k = 4$ permutations has some minor errors, and is repaired here. In addition, we state a lower bound and show a relationship to the feedback arc set problem. Finally, we establish a $2 - 2/k$ approximation, improving the previous factor of 2. The approximation is achieved by the best input permutation and is best possible for $k = 2$ and for the selection of the common permutation from the set of input permutations. The common rank aggregation methods minimize the sum of all disagreements over all permutations. Here we introduce the *maximum version*, $\text{PCM}_{\max}$-k, which expresses a fair aggregation and attempts to avoid a too severe discrimination of any participant or permutation. With the optimal solution, nobody should be totally unhappy. We show the NP-hardness of $\text{PCM}_{\max}$-k for all $k \geq 4$ and establish a 2-approximation, which is achieved by any input permutation. This parallels similar results for the Kemeny aggregation problem [1,8] and for the Coherence aggregation problem [4]. The case $\text{PCM}_{\max}$-2 with two permutations is efficiently solvable, whereas the case $k = 3$ remains open.

Besides the specific results, this work aims to bridge the gap between the combinatorics of rank aggregations and crossing minimizations in graph drawing, with a mutual exchange of notions, insights, and results. We can establish parallel results, but there is no direct transformation. This unification is an urgent open problem. In Section 2 we introduce the basic notions from graph drawing and rank aggregations, establish a lower bound and a relation to the feedback arc set problem, and show how to draw rank aggregations. In Section 3 we state the NP-hardness of the crossing minimization problem for just four permutations, prove the approximation results, and provide an ILP formulation. Finally, in Section 4 we investigate the special cases with two and three permutations.

2. Preliminaries

Given a set of alternatives U, a *ranking* π with respect to U is an ordering of a subset S of U such that $\pi = (x_1, x_2, \dots, x_r)$ with $x_i > x_{i+1}$, if x_i is ranked higher than x_{i+1} for some total order $>$ on U. For convenience, we assign unique integers to the items of U and let $U = \{1, \dots, n\}$. We call π a *(full) permutation*, if $S = U$, and a *partial permutation*, if $S \subseteq U$. A permutation is represented by an ordered list of items, where the rank of an item is given by its position in the ordered list, with the highest, most significant, or best item in first place. It is partial, if the items from $U - S$ are discarded from the ranking.

The rank aggregation or the *crossings of permutations problem* is to combine several rankings $\pi_1, \ldots, \pi_k$ on U, in order to obtain a common ranking π^*, which can be regarded as the compromise between the rankings. The goal is the best possible common ranking, where the notion of "best" depends on the objective. It is formally expressed as a cost measure or a penalty between the π_i and π^*; the *common* version takes the sum of the penalties, the *max* version is introduced here. Several of these criteria have a correspondence in graph drawing. A prominent and frequently studied criterion is the Kendall-τ distance [3,4,8,15]. The *Kendall-τ distance* of two permutations over $U = \{1, \ldots, n\}$ measures the number of pairwise disagreements or inversions, $K(\pi, \tau) = |\{(u, v) | \pi(u) < \pi(v) \text{ and } \tau(u) > \tau(v)\}|$. This value is invariant under renaming, or the application of a permutation σ on both π and τ; in particular we can assume without loss of generality that τ is the identity. For a set of permutations $P = \{\pi_1, \ldots, \pi_k\}$ the Kendall-τ distance generalizes by collecting all disagreements, $K(P, \pi^*) = \sum_{i=1}^{k} K(\pi_i, \pi^*)$. The value $K(P, \pi^*)$ can be expressed in various ways. For every pair of distinct items (u, v), the *agreement* $A_P(u, v)$ is the number of permutations from P which rank u higher than v, and the *disagreement* is $D_P(u, v) = k - A_P(u, v)$, where k is the number of permutations. Clearly, the agreement on (u, v) equals the disagreement on the reverse ordering (v, u). For every (unordered) pair of items, let $\Delta(u, v) = |k - 2A_P(u, v)|$ express the difference between the agreement and the disagreement of u and v.

There is an established lower bound for the Kendell-τ distance for the permutations of P, which is the sum over the least of the agreements and disagreements,

$$LB(P) = \sum_{u<v} \min\{A_P(u, v), D_P(u, v)\}.$$

Then the disagreement against a common permutation π^* is

$$K(P, \pi^*) = LB(P) + \sum_{\pi^*(u)<\pi^*(v) \text{ and } D_P(u,v)>A_P(u,v)} \Delta(u, v).$$

Thus $\Delta(u, v)$ is added as a penalty if π^* disagrees with the majority of the permutations. If there is a tie for the ranking of u and v in P, then just the term from the lower bound is taken into account.

Recall the crossing minimization problem of two-layered graphs, which is a major subproblem in the Sugiyama algorithm for drawings of directed graphs. The objective are "nice" drawings with few edge crossings and straight-line edges. Here the vertices of one layer are fixed, and the vertices on the other layer are free to be rearranged into any permutation. The disagreement (agreement) of two free vertices u and v is then the number of crossings of edges incident with u and v if u is placed left (right) of v. The lower bound $LB(P)$ from above gives a corresponding lower bound for the crossing minimization problem, which is often close to the optimum value, as established by experiments in [13].

Given a set of (full or partial) permutations $P = \{\pi_1, \ldots, \pi_k\}$ on a universe $U = \{1, \ldots, n\}$, the *crossing number* of P is the number of crossings against the best permutation π^* with respect to the Kendall-τ distance, i.e., $CR(P) = \min_{\pi^*} K(P, \pi^*)$. The *crossing minimization problem* is finding such a permutation π^*. We will refer to the crossing minimization problem of k permutations as the PCM-k problem. A new cost measure is the *max crossing number*, which attempts to minimize the number of crossings for any permutation. For a set of k permutations P and a target permutation π^* let $K_{\max}(P, \pi^*) = \max\{K(\pi_i, \pi^*) \mid \pi_i \in P\}$ and define the *max crossing number* of P by $CR_{\max}(P) = \min_{\pi^*} K_{\max}(P, \pi^*)$. The permutation π^* giving the value $CR_{\max}(P)$ is a solution to the max crossing minimization problem. This problem is referred to as the PCM$_{\max}$-k problem.

The crossing number represents an aggregation, which is the best compromise for the given lists of preferences and minimizes the number of disagreements. The objective behind the max crossing number is an arrangement that minimizes the average cost of each participant. This is different from the crossing number since the latter does not necessarily distribute the crossings evenly among the given permutations, and only the following obvious relationship between them is apparent:

Lemma 1. *For a set of k permutations $P = \{\pi_1, \ldots, \pi_k\}$,*

$$CR_{\max}(P) \leq CR(P) \leq k \cdot CR_{\max}(P).$$

Clearly, both objectives can be combined to the best possible permutation π^* which minimizes the sum of crossings and then balances their distribution. For $k = 2$, we will see that this gives the best possible bound of

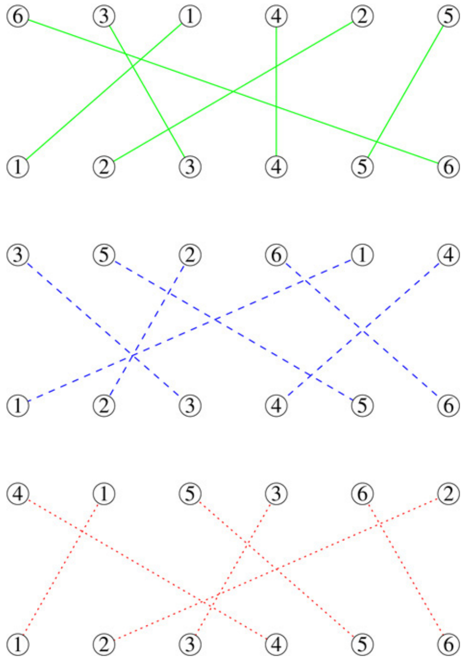

Fig. 1. Three permutations $\pi_1 = (6\,3\,1\,4\,2\,5)$ (green and solid), $\pi_2 = (3\,5\,2\,6\,1\,4)$ (blue and dashed), and $\pi_3 = (4\,1\,5\,3\,6\,2)$ (red and dotted). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

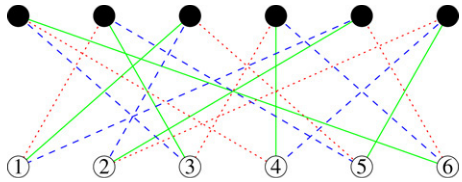

Fig. 2. Coloured permutation graph for $\pi_1 = (6\,3\,1\,4\,2\,5)$ (green and solid), $\pi_2 = (3\,5\,2\,6\,1\,4)$ (blue and dashed), and $\pi_3 = (4\,1\,5\,3\,6\,2)$ (red and dotted). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

$CR_{\max}(P) = \lceil CR(P)/k \rceil$. But for larger k this bound cannot always be achieved, as can be seen by letting P consist of a permutation over $1, \ldots, n$ and of $k-1$ copies of its reversal. Then $CR(P) \leq n(n-1)/2$ and $CR_{\max}(P) \geq n(n-1)/4$.

2.1. Drawing permutations

We now translate rank aggregations to graph drawing. Two permutations π and τ on a universe $U = \{1, \ldots, n\}$ are drawn as a two-layer bipartite graph with the vertices $1, \ldots, n$ on each layer in the order given by π and τ and a straight-line edge between the two occurrences of each item on the two layers. In partial permutations the vertices from $U - S$ are isolated. A set of k permutations $\pi_1, \ldots, \pi_k$ and a common permutation π^* are represented by a sequence of pairs of permutations, where the lower layer is fixed in all drawings. For convenience, we let the lower layer be the identity with $\pi^*(i) = i$, see Fig. 1.

Alternatively, we can merge the permutations into the *coloured permutation graph* G, which is a bipartite graph with k edge colours, such that there are vertices $1, \ldots, n$ on each layer. There is an edge in the ith colour between u on the upper layer and j on the lower layer if and only if $\pi_i(u) = j$. Every vertex has degree k with one edge in each colour if and only if the permutations are full. For the further discussion isolated vertices can be discarded and vertices of degree one can be ranked according to their permutation, since neither of them induces crossings. See also Fig. 2.

Obviously, for two full or partial permutations π and π^*, the Kendall-τ distance $K(\pi, \pi^*)$ is the number of edge crossings in a straight-line drawing of their bipartite graph. It ranges between 0 and $n(n-1)/2$ and can be efficiently computed either by accumulating for every i the number of items, which are greater than i and occur to the left of i in π, provided π^* is the identity, or by techniques from counting crossings in two-layer graphs in [21].

Lemma 2. *The Kendall-τ distance $K(\pi, \pi^*)$ of two (full or partial) permutations over $U = \{1, \ldots, n\}$ can be computed in $O(n \log n)$ time.*

The crossing number of a set of permutations $CR(\{\pi_1, \ldots, \pi_k\})$ is obtained by computing the best possible permutation π^* for the lower layer and then

(1) by counting and summing all crossings in the set of bipartite graphs for the permutations π_i and π^*, or
(2) by counting and summing all crossings of monochrome edges in the coloured permutation graph.

The max crossing number is obtained in the same drawings by taking the maximum number of crossings for each pair (π_i, π^*) or the maximum number of crossing for monochromatic edges in the coloured permutation graph.

Thus we have two alternative ways for a nice visual representation and we can measure the quality by "inspection". The translation makes techniques from graph drawing accessible to rank aggregation problems, e.g., Lemma 2. It helps in illustrating notions and gives hints towards useful concepts, e.g., the penalty graph and its use for lower bounds.

2.2. Penalty graphs

There is a direct relationship between the crossing minimization problem and the *feedback arc set problem*, which has been established at several places. Recall that the feedback arc set problem is finding the least number of arcs F in a directed graph $G = (V, E)$, such that every directed cycle contains at least one arc from F, i.e., the graph $G' = (V, E - F)$ is acyclic. In the more general weighted case, the objective is a set of arcs with least weight. In the two-layer crossing minimization problem, the *penalty graph* has arcs with weights corresponding to the difference between the number of crossings among the edges incident with two vertices u and v, if u is placed left of v, or vice versa. In their seminal paper, Sugiyama et al. [20] have introduced the penalty digraph for the two-layer crossing minimization problem, and in [2] it is used for voting tournaments. Demetrescu and Finocchi [5] have used this approach for the two-sided crossing minimization problem and have tested several heuristics. Recently, Ailon et al. [1] have established improved randomized approximations for aggregation and feedback arc set problems. For the crossing minimization problem for permutations, the penalty graph can be applied in the same spirit, but we use the difference in the majority counts $\Delta(u, v)$ as edge weights. Thus, for a set of permutations P over $\{1, \ldots, n\}$ the *penalty digraph* of P is a weighted directed graph $H = (V, A, w)$ with a vertex for each item u and an arc (u, v) with weight $\Delta(u, v)$ if and only if a strict majority of permutations rank u higher than v, i.e., if $(u-v)\cdot(D_P(u, v) - A_P(u, v)) < 0$. Let $w(FAS(P))$ denote the *weight* of the optimum feedback arc set in the penalty digraph.

First, we establish the connection between the crossing number and the feedback arc set of the penalty graph. For the two-layer crossing minimization problem it was first observed by Sugiyama [20], and used in various places, [1,5,9,16]. As a consequence, the crossing minimization problem can be reduced to a feedback arc set problem.

Theorem 3. *Let $P = \{\pi_1, \ldots, \pi_k\}$ be a set of permutations. Then the crossing number of P equals the lower bound plus the weight of the feedback arc set*

$$CR(P) = LB(P) + w(FAS(P)).$$

Proof. For any permutation π there are $LB(P)$ unavoidable inversions or crossings and $K(P, \pi) = LB(P) + \sum_{\pi(u)<\pi(v) \text{ and } D_P(u,v)>A_P(u,v)} \Delta(u, v)$. Now, the deletion of all arcs (u, v) with $u < v$ and $\pi(u) > \pi(v)$ from the penalty digraph of P leaves an acyclic digraph, since there are no cycles in a single permutation π. If π is such that $K(P, \pi)$ is minimal, then the set of arcs removed from the penalty graph is a feedback arc set.

Conversely, consider the penalty graph of P and remove any set of arcs F to make the remainder acyclic. Consider any permutation π which is in conformity with a topological ordering. Then $K(P, \pi) \leq LB(P) + \sum_{f \in F} \Delta(f)$, and if F is such that its weight is $w(FAS(P))$, then π is such that $K(P, \pi)$ is minimal. $\square$

The following property is obvious.

Corollary 4. *The penalty graph is acyclic iff the lower bound for the crossing minimization is the minimal bound.*

3. Complexity of optimal permutations

In this section we study the complexity of finding an optimal permutation for the common and the max crossing numbers. There are strong similarities to the one-sided crossing minimization problem, which go through to the number of permutations and the degrees of the free vertices. Crossing minimization in graphs is NP-hard. This holds true for general graphs [12], and even for two-layer graphs with the upper layer fixed. These graphs may be dense with about a third of all possible crossings [9] or sparse with degree $k = 4$ for the vertices on the free layer [16]. The case of degree 3-graphs for the free layer is still open. Correspondingly, there are NP-hardness results for permutations. For many partial permutations with just two elements the crossing minimization problem is in one-to-one correspondence with the feedback arc set problem, where every two element permutation represents an arc, and thus is NP-hard [10, 11]. By a different reduction from the feedback arc set problem, Bartholdi et al. [3] have proved the NP-hardness of Kemeny optimal permutations for many permutations. In [2] the first NP-hardness proof is credited to Orlin (1981, unpublished manuscript). A major strengthening has been claimed by Dwork [8] with a reduction from the feedback arc set problem to just four permutations. However, the construction in [8] needs some minor corrections.

Theorem 5. *The crossing minimization problem* PCM-*k is NP-hard for $k = 4$ full permutations. It remains NP-hard for $k \geq 6$ full permutations and k even, and for any $k \geq 4$ partial permutations.*

Proof. We first show how to reduce the feedback arc set problem to PCM-k for $k = 4$ full permutations. Then we extend the proof to the other cases with more permutations. Let $G = (V, E)$ be a directed graph with $V = \{v_1, \ldots, v_n\}$ and $|E| = m$ in which we want to find the smallest feedback arc set. For every vertex v let $out(v)$ be the sequence of outgoing edges in any order, and let $in(v)$ denote the sequence of incoming edges. Finally, for a sequence x let x^r denote its reversal, reading the elements right-to-left. Now, construct two pairs of full permutations from the vertices and edges of G.

$$\begin{aligned}
\pi_1 &= (v_1\ out(v_1) v_2\ out(v_2)\ \ldots\ v_n\ out(v_n)),\\
\pi_2 &= (v_n\ out(v_n)^r \ldots v_2\ out(v_2)^r\ v_1\ out(v_1)^r),\\
\pi_3 &= (in(v_1)\ v_1\ in(v_2)\ v_2 \ldots in(v_n)\ v_n), \quad \text{and}\\
\pi_4 &= (in(v_n)^r\ v_n \ldots in(v_2)^r\ v_2\ in(v_1)^r\ v_1).
\end{aligned}$$

Let $K' = 2\binom{n}{2} + 2\binom{m}{2} + 2m(n-1)$. The claim is now that G has a feedback set of size at most f iff $CR(P) \leq K = K' + 2f$.

In [8] the incoming edges are listed to the right of their vertices in π_3 and π_4, but then the construction does not work. And a different value for K is used. Consider an arbitrary permutation τ over $V \cup E$. For any pair of vertices u, v, regardless of the order of u and v in τ, there must be one crossing of u and v between τ and either π_1 or π_2 (but not both). Similarly, there must be one crossing in either π_3 or π_4 (but not both). Therefore τ creates $2\binom{n}{2}$ crossings among pairs of vertices. Similarly, τ creates $2\binom{m}{2}$ crossings among pairs of edges. Finally, consider a directed edge $e = (u, v)$. For any vertex $w \neq u$, there must be a crossing of e and w in π_1 or π_2 (but not both). For any vertex $w \neq v$, there must be a crossing of e and w in π_3 or π_4 (but not both). So τ has $2m(n-1)$ crossings among an edge and a vertex. Hence, τ must have at least K' crossings, and we have accounted for every possible crossing except for an edge with its tail (in π_1 and π_2) and an edge with its head (in π_3 and π_4.)

Let F be a feedback arc set, and consider the directed acyclic subgraph $G' = (V, E - F)$. Let $w_1, e_1, \ldots, e_{q_1}, w_2, e_{q_1+1}, \ldots, e_{q_2}, \ldots, w_n, e_{q_{n-1}+1}, \ldots, e_{q_n}$ be a topological sorting of the vertices and edges of G', where the topological order of u, e, and v is preserved for every directed edge $e = (u, v)$. Insert the edges from F into this sequence, say right of their vertices as in π_1 and π_2, and let π^* be the resulting permutation over the vertices and edges of G.

Then π^* has K' crossings from the edges of $E - F$. Consider the impact from the feedback arc set. If a feedback arc is placed as an outgoing arc as in the above π^*, then it induces two extra crossings with its incoming vertex in π_3 and π_4. In total these are $2|F|$. Hence, there are at most K crossings. Conversely, if there are at most K crossings in $\sum_{i=1}^{4} K(\pi_i, \pi^*)$, then K' crossings are unavoidable by the two pairs of permutations. Since all other pairs are completely taken, only crossings from pairs (v, e) can account to the extra up to $2|F|$ crossings, where v and e are incident, i.e., $e = (u, v)$ or $e = (v, u)$. If an edge $e = (u, v)$ induces a crossing with u in (π_1, π^*), it induces a second crossing in (π_2, π^*), and vice versa. Thus it contributes 0 or 2 crossings. This holds accordingly for e and v and π_3

and π_4. Let F be this set of edges. If we delete the edges from F, then the remaining edges respect the ordering in π^*, and the subgraph $G' = (V, E - F)$ is acyclic.

This completes the NP-hardness proof with $k = 4$ full permutations. For $k = 6, 8, \ldots$ we add $(k-4)/2$ pairs of full permutations (π, π^r), where π^r is the reversal of π. Each such pair causes one crossing per item and increases the number of crossings by $(n+m)(n+m-1)/2 \cdot (k-4)/2$, as already explained in [8]. This technique does not work for odd numbers of full permutations, where the complexity remains open. However, we may add any number of primitive partial permutations over singletons, which do not increase the number of crossings. Since every full permutation is a partial permutation, this solves the case with $k \geq 4$ partial permutations, four of which may be full permutations. □

For the common crossing minimization problem we sum the number of crossings in the sequence of bipartite graphs or of monochrome edges. In the max problem we wish to minimize the maximal number of such crossings, i.e., we wish to treat every arrangement as fair as possible.

Theorem 6. *The max crossing minimization problem* PCM$_{\max}$-*k is NP-hard for any* $k \geq 4$ *permutations.*

Proof. We reduce from the common crossing minimization problem for four permutations, see Theorem 5. Let $P = \{\pi_1, \ldots, \pi_4\}$ be the four permutations among which we want to minimize the sum of crossings. We construct a set Q of four new permutations by applying $\pi_1, \ldots, \pi_4$ to four sets of n elements, say $[1 \ldots n]$, $[n+1 \ldots 2n]$, $[2n+1 \ldots 3n]$, and $[3n+1 \ldots 4n]$ as follows:

$$\begin{aligned}
\sigma_1 &= (\pi_1[1 \ldots n]\ \pi_2[n+1 \ldots 2n]\ \pi_3[2n+1 \ldots 3n]\ \pi_4[3n+1 \ldots 4n]),\\
\sigma_2 &= (\pi_2[1 \ldots n]\ \pi_3[n+1 \ldots 2n]\ \pi_4[2n+1 \ldots 3n]\ \pi_1[3n+1 \ldots 4n]),\\
\sigma_3 &= (\pi_3[1 \ldots n]\ \pi_4[n+1 \ldots 2n]\ \pi_1[2n+1 \ldots 3n]\ \pi_2[3n+1 \ldots 4n]),\\
\sigma_4 &= (\pi_4[1 \ldots n]\ \pi_1[n+1 \ldots 2n]\ \pi_2[2n+1 \ldots 3n]\ \pi_3[3n+1 \ldots 4n]).
\end{aligned}$$

We claim that we can recover the optimal permutation for $CR(P)$ from the optimal solution for $CR_{\max}(Q)$, and to do so, we study the common minimization problem for Q, i.e., $CR(Q)$. Let σ^* be the optimal solution for $CR(Q)$. It must consist of four blocks for the four disjoint sets of n elements, since a rearrangement of the elements from disjoint sets decreases the number of crossings. So

$$\sigma^* = (\pi_1^*[1 \ldots n]\ \pi_2^*[n+1 \ldots 2n]\ \pi_3^*[2n+1 \ldots 3n]\ \pi_4^*[3n+1 \ldots 4n]).$$

Note that each block hence contributes crossings to one copy of each permutation in P. In particular, each π_i^* must be optimal for $CR(P)$ (otherwise we could improve σ^*.) Hence, we may assume that $\pi_1^* = \pi_2^* = \pi_3^* = \pi_4^* = \pi^*$ for some permutation π^* that is optimal for $CR(P)$. Now for each i, the distance between σ^* and σ_i is exactly $K(P, \pi^*)$, and in particular, this distance is equal for all i. So $CR_{\max}(Q) \leq \max_i K(\sigma_i, \sigma^*) = K(P, \pi^*) = \frac{1}{4} CR(Q)$. By Lemma 1 equality holds, so we can recover $CR(P)$ from $CR_{\max}(Q)$ (and the optimal permutation by taking any of the permutations within each block.) This finishes the proof for $k = 4$. For $k \geq 5$, observe that the maximum number of crossings does not change, if the first permutation σ_1 is also taken as the 5th, 6th, ... permutations. Note that any full permutation is also a partial permutation. So the NP-hardness holds whether or not partial permutations are allowed. □

3.1. Approximation algorithms

Since the crossing minimization problems are NP-hard for any (even) $k \geq 4$, we cannot hope to find the best solution in polynomial time, and hence study other ways to attack the problem. One way is to consider approximation algorithms, which we study next. There is a close connection between the number of crossings, i.e., the Kendall-τ distance and the Spearman footrule distance, as established in [6]. The *Spearman footrule distance* accumulates the linear arrangement or the length between two permutations over $\{1, \ldots, n\}$ by $f(\pi, \tau) = \sum_{i=1}^{n} |\pi(i) - \tau(i)|$. Again this extends to a set P of permutations in the common version by summation $f(P, \pi^*) = \sum_{j=1}^{k} f(\pi_j, \pi^*)$, and in the max version $f_{\max}(P, \pi^*) = \max_j f(\pi_j, \pi^*)$.

For a pair of permutations, every move induces a disarrangement and each crossing implies that at most two elements must move each by one position. Hence, $K(\pi, \tau) \leq f(\pi, \tau) \leq 2K(\pi, \tau)$ for full permutations π and τ, as established in [6]. If π^* and $\hat{\pi}$ are the optimal permutations for the Kendall-τ and the Spearman footrule distances, respectively, then $K(P) = K(P, \pi^*) \le K(P, \hat{\pi}) \le f(P, \hat{\pi}) = f(P)$, and, accordingly, $f(P) \le 2K(P)$ and $K_{\max}(P) \le f_{\max}(P) \le 2K_{\max}(P)$. The optimal permutation for the Spearman footrule distance can be computed by solving a weighted perfect bipartite matching problem, as explained in [8], with n nodes on either side and weights $w(i, j) = \sum_{i=1}^{n} |\pi(i) - j|$ for $1 \le i, j \le n$. Hence, there is a 2-approximation for $K(P)$.

An alternative 2-approximation for the Kendall-τ distance is obtained by choosing the best among the given permutations, see [1], and there is a simple 2-approximation for the coherence complexity [4]. We now show that the technique of choosing the best among the given permutations in fact gives an even better approximation, in particular for small values of k.

Theorem 7. *There is a $(2 - \frac{2}{k})$-approximation for the (common) crossing minimization problem* PCM-k.

Proof. We claim that the best permutation among $\pi_1, \dots, \pi_k$ is a $(2-2/k)$-approximation to the optimal permutation, and show this for odd k as follows. Let $P = \pi_1, \dots, \pi_k$ be the input permutations. For $a > d$ and $a + d = k$, let $E_{a,d}$ be those arcs (u, v) for which $A_P(u, v) = a$ and $D_P(u, v) = d$, i.e., u comes before v in a permutations, and after v in d permutations. Denote $m_{a,d} = |E_{a,d}|$. Consider the k vertex orderings defined by the k permutations, and count the number of arcs that are reversed in them. For $a > d$, each arc in $E_{a,d}$ must be reversed in exactly d of the permutations, hence the total number of reversed arcs is

$$L = m_{k-1,1} + 2m_{k-2,2} + \cdots + jm_{k-j,j} = \sum_{a>d,a+d=k} dm_{a,d}, \tag{1}$$

where $j = \lceil (k-1)/2 \rceil$. By the pigeon hole principle, therefore in at least one of the permutations (say in π_1), the number of reversed arcs is at most $1/k$th of Eq. (1). Denote by $r_{a,d}$ the number of arcs in $E_{a,d}$ that are reversed in π_1, then we therefore have

$$r_{k-1,1} + r_{k-2,2} + \cdots + r_{k-j,j} \le \frac{1}{k}\left(m_{k-1,1} + 2m_{k-2,2} + \cdots + jm_{k-j,j}\right).$$

Each arc in $E_{a,d}$ has weight $a - d$ in the feedback arc set problem, so the weight of the feedback arc set solution defined by π_1 is

$$\begin{aligned} w(FAS) &= (k-2)r_{k-1,1} + (k-4)r_{k-2,2} + \cdots + (k-2j)r_{k-j,j} \\ &\le (k-2)r_{k-1,1} + (k-2)r_{k-2,2} + \cdots + (k-2)r_{k-j,j} \\ &\le (k-2)\frac{1}{k}\left(m_{k-1,1} + 2m_{k-2,2} + \cdots + jm_{k-j,j}\right) = \frac{k-2}{k}L. \end{aligned}$$

Now note that L of Eq. (1) also exactly equals the lower bound $LB(P)$, since we only consider edges in $E_{a,d}$ with $a > d$. Therefore, the number of crossings obtained with π_1 is

$$LB(P) + w(FAS) \le L + \frac{k-2}{k}L = \left(2 - \frac{2}{k}\right)L \le \left(2 - \frac{2}{k}\right)\mathrm{OPT},$$

where OPT is the number of crossings in the optimal solution. This finishes the proof for odd k. The proof for even k is almost identical, but we must be careful for the case $a = d = k/2$. Here every arc in $E_{k/2,k/2}$ also has the reverse arc in $E_{k/2,k/2}$, leading to a double-counting. The proof goes through if for each vertex-pair u, v with $A_p(u, v) = D_P(u, v) = k/2$, we include only one of the directed edges (u, v) and (v, u) in $E_{k/2,k/2}$. □

The previous bound is optimal in some cases. This clearly holds for the case of just two permutations, which is discussed below. Moreover, we note here that if the target permutation is taken from the given set of permutations, the $(2 - \frac{2}{k})$-approximation is best possible for PCM-k. Namely, let $\sigma_1, \dots, \sigma_k$ be k permutations over k copies of pairwise disjoint elements, and consider the following k permutations (over k distinct sets of n elements; we will not specifically list these here):

$$\begin{aligned} \pi_1 &= (\sigma_1^r \sigma_2 \sigma_3 \dots \sigma_k) \\ \pi_2 &= (\sigma_1 \sigma_2^r \sigma_3 \dots \sigma_k) \\ \pi_3 &= (\sigma_1 \sigma_2 \sigma_3^r \dots \sigma_k) \end{aligned}$$

$$\vdots \qquad \vdots$$
$$\pi_k = (\sigma_1\sigma_2\sigma_3 \dots \sigma_k^r).$$

Then $\pi^* = (\sigma_1\sigma_2 \dots \sigma_k)$ achieves $k\binom{N}{2}$ crossings. However, any π_i disagrees with any π_j on the directions of both σ_i and σ_j, and hence creates $2(k-1)\binom{N}{2}$ crossings, which is $\frac{2k-2}{k} = 2 - \frac{2}{k}$ times the optimum.

Now we turn to approximation algorithms for the max version of the problem. Here, choosing any of the input permutations yields a 2-approximation, and again, this is optimal for the given set of permutations.

Theorem 8. *There is a* 2*-approximation for the max crossing minimization problem* $\mathrm{PCM}_{\max}$-k.

Proof. Let $\pi_1, \dots, \pi_k$ be a given set of permutations. We claim that any of these permutations is a 2-approximation, and prove this for π_1. Let π^* be the optimal permutation for the $\mathrm{PCM}_{\max}$-k problem, and let j^* be the index of the permutation where the maximum is achieved in the optimal solution, i.e.,

$$K(\pi_{j^*}, \pi^*) \geq K(\pi_i, \pi^*) \quad \text{for all } i.$$

Note that the optimal value OPT equals therefore $K(\pi_{j^*}, \pi^*)$. Now for any permutation π_i, we have

$$K(\pi_i, \pi_1) \leq K(\pi_i, \pi^*) + K(\pi^*, \pi_1) \leq K(\pi_{j^*}, \pi^*) + K(\pi^*, \pi_{j^*}) = 2\mathrm{OPT},$$

so $\max_i K(\pi_i, \pi_1) \leq 2\mathrm{OPT}$, and therefore π_1 is a 2-approximation for the max crossing number problem. □

As above, if the target permutation is taken from the given set of permutations, the 2-approximation is best possible for $\mathrm{PCM}_{\max}$-k. To see this use any permutation π and its reversal π^r. Then $CR(\pi, \pi^r) = n(n-1)/2$ and $CR_{\max}(\pi, \pi^r) = \lceil n(n-1)/4 \rceil$.

3.2. Other remarks on approximation algorithms

The approximation algorithms given above are quite straightforward: Simply testing all input permutations, and picking the best one among them gives a 2-approximation (for both versions of the problem). While this gives the best bound if we only are allowed to pick among the input permutations (as we showed), there is no particular reason why one should be allowed to pick only among the input permutations. So finding a better approximation algorithm that truly picks a permutation "between" the input permutations is an urgent open problem. Unfortunately, we have not been able to develop a better strategy. We mention here some related results that may lead to better approximation algorithms. We have not been able to prove better theoretical bounds for these, and doing experiments to see whether they work well in practice remains future work.

- Using standard LP-techniques we can formulate both PCM-k and $\mathrm{PCM}_{\max}$-k as an integer linear program using $O(n^4 + k)$ variables and constraints. The main idea is to use indicator variables $s_{i,j,k,l} \in \{0, 1\}$ which are 1 if and only if $\pi^*(i) = j$ and $\pi^*(k) = l$; we can then express the number of crossings in terms of the $s_{i,j,k,l}$ and minimize their sum or their maximum. Using such a formulation, we can solve the problem exactly (but not in polynomial time) using any ILP solver. But more interesting would be to test whether approximations can be obtained this way. The standard technique is to solve the fractional relaxation of the ILP and apply rounding. Can this be used for good approximation algorithms?
- In an interesting parallel, the best approximation bound for one-sided two-layer crossing minimization long stood at 2 as well [22], but was recently improved to 1.4664 [18]. Some randomized approximations have been established in [1]. Neither of these results seems easily transferable to PCM-k or $\mathrm{PCM}_{\max}$-k for a theoretical bound. How well do these techniques work in experiments?

4. The small cases

We now consider PCM-k and $\mathrm{PCM}_{\max}$-k for small values of k. Clearly, for $k = 1$, a single user will take his preferences for the optimal arrangement, and then there are no crossings. Consider the case $k = 2$. For bipartite graphs with vertices of degree 2 on the lower layer the one-sided crossing minimization problem is solvable in linear time by the barycenter heuristic. This holds because the nesting structure of the neighbours on the upper layer determines

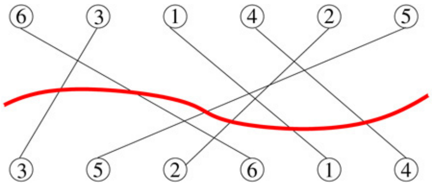

Fig. 3. Crossings for 2 permutations.

the left–right positions in an optimal layout, see [16]. The main ingredient here is that the penalty digraph is acyclic. The permutation crossing number can be found easily for two permutations π_1 and π_2; π_1 itself is optimal with value $c = K(\pi_1, \pi_2)$. Many optimal permutations can be found from a straight-line drawing of π_1 and π_2, see also Fig. 3. Consider an arbitrary curve from left to right that crosses each straight line (v, v) for $v = 1, \ldots, n$ exactly once (we call such a curve a *pseudo-line*.) This yields a permutation π^* by listing the elements in the order in which they were crossed. Any permutation obtained in such a way is optimal for PCM-2. For example, for $\pi_1 = (6\,3\,1\,4\,2\,5)$ and $\pi_2 = (3\,5\,2\,6\,1\,4)$, π_1 and π_2 themselves and also $(3\,6\,5\,2\,1\,4)$ are optimal, see Fig. 3.

Using these "intermediate" permutations, the max crossing problem can be solved in polynomial time by a sweep-line technique [19]. Since the sum of the number of crossings c is determined, the max crossing minimization problem is solved by distributing these crossings uniformly to either side such that $CR_{\max}(\pi_1, \pi_2) = \lceil c/2 \rceil$. An optimal permutation – which is best possible both for the sum and for the maximum – can be computed in $O(n + r) \log n$ time by a standard sweep-line technique, where r is the number of crossings. We summarize:

Theorem 9. PCM-2 *can be solved in* $O(1)$ *time, and* PCM$_{\max}$*-2 takes at most* $O(n^2 \log n)$ *time.*

Now we address the case $k = 3$. Here, the complexity is open, both for permutations and for one-sided two-layered graphs with degree k on the free layer [16].

For the crossings of permutations problem the case with odd numbers is special. For every pair of items u and v there is a clear winner. There are no ties and the penalty graph is a complete tournament, i.e., there is exactly one directed arc (u, v) or (v, u) between each pair of vertices with an odd weight between 1 and k. Then every cycle c has a 3-cycle, which is a subcycle of length three [17]. There are simple permutations including a cycle, e.g. (1, 2, 3), (2, 3, 1) and (3, 2, 1). For three permutations Theorem 7 gives a 4/3-bound on the approximation. We can show that this is not tight, and give some insights into how it could be improved, in the following. In the penalty graph there are 3-edges with weight 3 from an unanimous decision of the three voters for two items, and 1-edges from a 2 : 1 decision. 3-edges cannot occur in 3-cycles, because for any 3-edge (u, v) and any other edge (v, t), the edge between u and t must be directed (u, t) as well. Similarly and edge (s, u) implies (s, v). Let π^* be an optimal permutation and let G^* be the associated penalty graph with the vertices given by their numbers in π^*. We claim that any 3-edge (u, w) is a forward-edge in the penalty graph (i.e., $u < w$), for otherwise their swap decreases the cost. To see this, assume that $w < u$ and consider any v between w and u (all others are not affected by the swap). Since edges from v to both u, w exist, and (u, w) is not in a directed 3-cycle, either edge (v, w) or edge (u, v) (or both) must have this direction. Hence, reversing the places of u and w does not increase the cost of edges incident to v, and decreases the cost by at least 3. Moreover, there is a backedge e in G^* iff there is a 3-cycle through e (otherwise a similar argument shows that exchanging the endpoints of e decreases the cost). Suppose there are at least p 3-cycles in G^* with mutually disjoint edges. A bound for p can be computed by a greedy algorithm checking the edges in increasing order of their occurrences in such subcycles. This p is a lower bound for the feedback arc set problem of G^*, since each such subcycle must be destroyed. Note that this lower bound is in *addition* to the lower bound $LB(P)$ that arises from the agreements and disagreements. If $p = 0$, then G^* is acyclic and the optimal solution can be computed easily. If $p > 0$, then the best input permutation has at most $4/3\ LB(P)$ crossings (see the proof of Theorem 7); by $p > 0$ it is therefore a q-approximation with $q < 4/3$. We conjecture that an even better approximation bound should be achievable here: If p is small, then the graph is almost acyclic and a greedy strategy for the feedback arc set should do well. If p is large, then the lower bound is large and simply choosing an input permutation should give a better (theoretical) approximation bound.

5. Conclusion

In this paper, we investigated the problem of rank aggregation, which corresponds to finding a permutation that minimizes the number of crossings with a given set of permutations. We introduced a variant that instead considers the maximum number of crossings among those permutations. We investigated complexity results and approximation algorithms. This problem is a one-sided two-layer crossing minimization problem in an edge-coloured bipartite graph, where only crossings between equally coloured edges are counted. As such, it is not surprising that the complexity results for our problem mirror the ones for one-sided two-layer crossing minimization. We end by mentioning some of the numerous open problems that remain in this field:

(1) How do the common techniques from one-sided two-layer crossing minimization, such as barycenter and median heuristics, sifting, or ILP approaches perform for the crossing minimization of permutations?
(2) Improve the approximations and establish bounds for partial permutations. The NP-hardness result for PCM-k only holds for k even. Could we improve the approximation results at least for k odd?
(3) The case $k = 3$ remains wide open. Is it NP-hard or polynomial?

Acknowledgments

The authors would like to thank Wolfgang Brunner, Christof König and Marcus Raitner for inspiring discussions, and the anonymous referees for the useful comments.

References

[1] N. Ailon, M. Charikar, A. Newman, Aggregating inconsistent information: Ranking and clustering, STOC (2005) 684–693.
[2] J.P. Barthelemy, A. Guenoche, O. Hudry, Median linear orders: Heuristics and a branch and bound algorithm, Europ. J. Oper. Res. 42 (1989) 313–325.
[3] J. Bartholdi III, C.A. Tovey, M.A. Trick, Voting schemes for which it can be difficult to tell who won the election, Soc. Choice Welfare 6 (1989) 157–165.
[4] F.Y.L. Chin, X. Deng, Q. Feng, S. Zhu, Approximate and dynamic rank aggregation, Theoret. Comput. Sci. 325 (2004) 409–424.
[5] C. Demetrescu, I. Finocchi, Breaking cycles for minimizing crossings, Electronic J. Algorithm Engineering 6 (2) (2001).
[6] P. Diaconis, R. Graham, Spearman's footrule as a measure for disarray, J. Royal Statistical Society, Series B 39 (1977) 262–268.
[7] G. Di Battista, P. Eades, R. Tamassia, I.G. Tollis, Graph Drawing: Algorithms for the Visualization of Graphs, Prentice Hall, 1999.
[8] C. Dwork, R. Kumar, M. Noar, D. Sivakumar, Rank aggregation methods for the Web, Proc. WWW10 (2001) 613–622.
[9] P. Eades, N.C. Wormald, Edge crossings in drawings of bipartite graphs, Algorithmica 11 (1994) 379–403.
[10] G. Even, J. Naor, B. Schieber, M. Sudan, Approximating minimum feedback sets and multicuts in directed graphs, Algorithmica 20 (1998) 151–174.
[11] M.R. Garey, D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman, San Francisco, 1979.
[12] M.R. Garey, D.S. Johnson, Crossing number is NP-complete, SIAM J. Alg. Disc. Meth. 4 (1983) 312–316.
[13] M. Jünger, P. Mutzel, 2-layer straightline crossing minimization: Performance of exact and heuristic algorithms, J. Graph Alg. Appl. 1 (1997) 1–25.
[14] M. Kaufmann, D. Wagner (Eds.), Drawing Graphs: Methods and Models, in: LNCS, vol. 2025, 2001.
[15] J.G. Kemeny, Mathematics without numbers, Daedalus 88 (1959) 577–591.
[16] X. Munos, W. Unger, I. Vrto, One sided crossing minimization is NP-hard for sparse graphs, in: Proc. GD 2001, in: LNCS, vol. 2265, 2002, pp. 115–123.
[17] J.W. Moon, Topics on Tournaments, Holt, New York, 1968.
[18] H. Nagamochi, An improved approximation to the one-sided bilayer drawing, Discrete Comput. Geometry 33 (4) (2005) 569–591.
[19] F.P. Preparata, M.I. Shamos, Computational Geometry: An Introduction, Springer-Verlag, Heidelberg, 1985.
[20] K. Sugiyama, S. Tagawa, M. Toda, Methods for visual understanding of hierarchical systems structures, IEEE Trans. SMC 11 (1981) 109–125.
[21] V. Waddle, A. Malhotra, An $E \log E$ line crossing algorithm for leveled graphs, in: Proc. GD 99, in: LNCS, vol. 1731, 2000, pp. 59–70.
[22] A. Yamaguchi, A. Sugimoto, An approximation algorithm for the two-layered graph drawing problem, Discrete Comput. Geom. 33 (2005) 565–591.