Electronic Colloquium on Computational Complexity, Report No. 131 (2005)

Ordering by weighted number of wins gives a good ranking for weighted tournaments*

Don Coppersmith† Lisa Fleischer‡ Atri Rudra§

Abstract

We consider the following simple algorithm for feedback arc set problem in weighted tournaments — order the vertices by their weighted indegrees. We show that this algorithm has an approximation guarantee of 5 if the weights satisfy *probability constraints* (for any pair of vertices u and v, $w_{uv} + w_{vu} = 1$). Special cases of feedback arc set problem in such weighted tournaments include feedback arc set problem in unweighted tournaments and rank aggregation. Finally, for any constant $\epsilon > 0$, we exhibit an infinite family of (unweighted) tournaments for which the above algorithm (*irrespective* of how ties are broken) has an approximation ratio of $5 - \epsilon$.

1 Introduction

Consider a sports tournament where all n players play each other and after all the $\binom{n}{2}$ games are completed, one would like to rank the players with as few inconsistencies as possible. By an inconsistency we mean a higher ranked player actually lost to a lower ranked player. A natural way to generate such a ranking is to rank according to the number of wins (with ties broken in some manner). We show that this natural heuristic has a provably good performance guarantee.

A weighted tournament with *probability constraints* is a complete directed graph $T = (V, E, w)$ where $w_{(\cdot)}$ is the weight function such that for any $u, v \in V$ with $u \neq v$, $w_{uv} + w_{vu} = 1$ and $w_{uv}, w_{vu} \geq 0$. We will use the term tournament to refer to a weighted tournament with probability constraints. An *unweighted* tournament is a special case of weighted tournaments with probability constraints, where the weights of the edges are either 0 or 1. The *minimum feedback arc set* in T is the smallest weight set $E' \subseteq E$ such that $(V, E \setminus E')$ is acyclic. Alternatively, a minimum feedback arc set can be described by an ordering $\sigma : V \to \{0, 1, \cdots, n-1\}$ which minimizes the weight the of *backedges* induced by σ where a backedge $(u, v) \in E$ satisfies $\sigma(u) > \sigma(n)$.

*This work will appear in Proceeding of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA06).

†Center for Communication Research, IDA, Princeton, NJ 08540. This work was done while the author was at IBM TJ Watson Research Center, Yorktown Heights, NY.

‡IBM TJ Watson Research Center, Yorktown Heights, NY 10598. *Email :* lkf@us.ibm.com. Supported in part by NSF grant CCF-0515127.

§Department of Computer Science and Engineering, University of Washington, Seattle, WA 98195. *Email :* atri@cs.washington.edu. This work was done while the author was a summer intern at IBM TJ Watson Research Center, Yorktown Heights, NY.

ISSN 1433-8092

The feedback arc set problem in general directed graphs can be approximated to within $O(\log n \log\log n)$ [12, 17] and is APX-Hard [15, 10]. The complementary problem of the maximum acyclic subgraph problem[1] can be approximated to within $2/(1+\Omega(1/\sqrt{\Delta}))$ (where Δ is the maximum degree) [6, 14] and is APX-Hard [16]. The feedback arc set problem in tournaments (shortened to FAS-TOURNAMENT for the rest of the paper) was conjectured to be NP-hard for a long time [4]. The conjecture was very recently proved in [1, 2][2]. The work of Ailon, Charikar and Newman ([1]) also describe the following simple randomized 3-approximation algorithm for *unweighted* FAS-TOURNAMENT. Their algorithm first picks a random vertex p to be the "pivot" vertex. All the vertices which are connected to p with an out-edge are placed to the "left" of p and the vertices which are connected to p through an in-edge are placed to the "right". Then, the algorithm recurses on the two tournaments induced by the vertices placed on either side of p.

Weighted FAS-TOURNAMENT is defined on a weighted tournament T where the weights satisfy probability constraints. Ailon et al. show that running their algorithm for FAS-TOURNAMENT on the unweighted tournament that is the weighted majority[3] of T yields a 5 approximation for weighted FAS-TOURNAMENT when the weights satisfy probability constraints.

There is a much simpler algorithm for both weighted and unweighted FAS-TOURNAMENT than ones considered by Ailon et al. — order the vertices in increasing order of their (weighted) indegrees (ties are broken arbitrarily). We analyze this algorithm (which we call INCR-INDEG in this paper) and show that it has an approximation guarantee of 5 for both unweighted FAS-TOURNAMENT and weighted FAS-TOURNAMENT when weights satisfy probability constraints.

We also study the problem of RANK-AGGREGATION. In this problem, given n candidates and k permutations of the candidates $\{\pi_1, \pi_2, \cdots, \pi_k\}$, we need to find the *Kemeny optimal* ranking, that is, a ranking π such that $\kappa(\pi, \pi_1, \pi_2, \cdots, \pi_k) = \sum_{i=1}^{k} \mathcal{K}(\pi, \pi_i)$ is minimized, where $\mathcal{K}(\pi_i, \pi_j)$ denotes the number of pairs of candidates that are ranked differently by π_i and π_j. The problem is NP-hard even when the number of lists is only four [5, 11]. There is a simple deterministic 2-approximation for this problem — pick the best of the input rankings. Ailon et al. reduce RANK-AGGREGATION to weighted FAS-TOURNAMENT with probability constraints [1]. This reduction implies that INCR-INDEG is a 5-approximation for RANK-AGGREGATION. Interestingly, INCR-INDEG in the RANK-AGGREGATION setting is exactly the same as the Borda's method [7]. Thus, our results show that the Borda's method is a factor 5 approximation of the Kemeny optimal ranking. Fagin et al. also show that the ranking produced by Borda's method is within a constant factor of the Kemeny optimal ranking [13].

Finally, for any $\epsilon > 0$, we exhibit an infinite family of (unweighted) tournaments for which INCR-INDEG has an approximation ratio of $5 - \epsilon$ (*irrespective* of how ties are broken). This result shows that our analysis is tight. This is somewhat surprising as it shows that the "dumbest" way of breaking ties is as good as the best tie breaking mechanism (at least from the viewpoint of approximation guarantees).

Independent of our work, van Zuylen [18] has designed a deterministic algorithm for FAS-TOURNAMENT with an approximation guarantee of 3 (when the weights satisfy probability constraints). Her algorithm derandomizes the algorithm of Ailon et al. mentioned earlier in this

[1]The maximum acyclic subgraph of a directed graph $G = (V, E)$ is the largest cardinality subset $E' \subseteq E$ such that the graph (V, E') is acyclic.

[2]Ailon, Charikar and Newman showed that the problem is NP-hard under randomized reductions [1]. Alon derandomized their construction [2]. A simpler reduction has been obtained by Contizer [8].

[3]The weighted majority of a weighted tournament T is defined as follows. For any pairs of vertices u and v, orient the edge between u and v in the direction which has the larger weight (breaking ties arbitrarily).

section (the pivot is chosen based upon the solution to the relaxation of a well known LP for FAS-TOURNAMENT).

The rest of the paper is organized as follows. We introduce some notation and some known facts in Section 2. We analyze the algorithm INCR-INDEG in Section 3. In Section 4, we present an infinite family of tournaments for which INCR-INDEG has an approximation factor of $5 - \epsilon$, for any $\epsilon > 0$. Finally, we conclude with some open questions in Section 5.

2 Preliminaries

We first fix some notations. For any positive integer m we will use $[m]$ to denote the set $\{0, 1, \cdots, m-1\}$. Also for any pairs of integers $a < b$, we will use $[a, b]$ to denote the set $\{a, a+1, \cdots, b\}$. We will also use $(a, b]$ to denote the set $\{a+1, \cdots, b\}$. The vertex set of the input tournament is assumed to be $[n]$. For any edge (u, v) in T, its weight is given by $w_{uv} \geq 0$. For the rest of the paper, the weights are assumed to satisfy *probability constraints*, that is, for any $u, v \in [n]$, the following identity holds — $w_{uv} + w_{vu} = 1$. For any vertex $v \in [n]$, $\texttt{In}(v)$ denotes the (weighted) indegree of v, that is,

$$\texttt{In}(v) = \sum_{u \in [n] \setminus \{v\}} w_{uv}.$$

We will use $\sigma : [n] \to [n]$ as a generic permutation. $\mathcal{O}$ will denote the permutation returned by the optimal algorithm for FAS-TOURNAMENT on input T while $\mathcal{A}$ will denote the permutation returned by INCR-INDEG. Given any permutation σ, B_σ denotes the sum of the weight of backedges induced by σ on T, that is,

$$B_\sigma = \sum_{u,v \in [n] : \sigma(u) > \sigma(v)} w_{uv}.$$

We now recall two well known notions of distances between permutations. Given permutations $\sigma, \rho : [n] \to [n]$; the *Spearmans' footrule distance* between the two permutations is given by

$$\mathcal{F}(\sigma, \rho) = \sum_{v \in [n]} |\sigma(v) - \rho(v)|$$

and the *Kendall-Tau distance* is given by

$$\mathcal{K}(\sigma, \rho) = \frac{1}{2} \cdot \sum_{u,v \in [n]} \mathbf{1}_{(\sigma(u) - \sigma(v)) \cdot (\rho(u) - \rho(v)) < 0}$$

where $\mathbf{1}_{(\cdot)}$ is the indicator function. In other words, $\mathcal{K}(\sigma, \rho)$ is the number of (ordered) pairs which are ordered differently by σ and ρ. The following relationship was shown in [9].

$$\mathcal{K}(\sigma, \rho) \leq \mathcal{F}(\sigma, \rho) \leq 2\mathcal{K}(\sigma, \rho) \tag{1}$$

For RANK-AGGREGATION, we will need the notion of *Kemeny distance*. Given the input lists $\{\pi_1, \pi_2, \cdots, \pi_k\}$ and an aggregate permutation σ, the Kemeny distance is defined as

$$\kappa(\sigma, \pi_1, \pi_2, \cdots, \pi_k) = \sum_{i=1}^{k} \mathcal{K}(\sigma, \pi_i)$$

We now present the reduction of RANK-AGGREGATION to weighted FAS-TOURNAMENT with probability constraints from [1]. Let $\{\pi_1, \cdots, \pi_k\}$ be a RANK-AGGREGATION instance on the set of candidates $[n]$. The equivalent weighted FAS-TOURNAMENT instance is a weighted tournament on $[n]$ such that for any pair of vertices i and j, w_{ij} is the fraction of input permutations which rank i before j. Note that by construction the weights satisfy probability constraints. Finally, sorting the vertices by their weighted indegrees (on the weighted tournament constructed above), is the well known Borda's method [7].

3 The algorithm for FAS-TOURNAMENT

We will analyze INCR-INDEG in this section. Recall that INCR-INDEG orders the vertices by their (weighted) indegrees (with ties being broken arbitrarily).

Our main result is the following theorem.

Theorem 1 INCR-INDEG *is a* 5*-approximation for weighted* FAS-TOURNAMENT*, that is,* $B_{\mathcal{A}} \leq 5B_{\mathcal{O}}$.

The reduction from FAS-TOURNAMENT to RANK-AGGREGATION outlined in Section 2 implies the following corollaries to Theorem 1.

Corollary 1 INCR-INDEG *is a* 5*-approximation for* RANK-AGGREGATION.

Corollary 2 *If* σ *is the Kemeny optimal rank aggregation for input rankings* $\pi_1.\pi_2, \cdots, \pi_k$ *and* σ' *is an output of the Borda's method for the same input, then*

$$\kappa(\sigma', \pi_1, \pi_2, \cdots, \pi_k) \leq 5\kappa(\sigma, \pi_1, \pi_2, \cdots, \pi_k).$$

We will prove Theorem 1 through a sequence of lemmas.

Lemma 1 *For any permutation* $\sigma : [n] \to [n]$,

$$2B_\sigma \geq \sum_{v\in[n]} |\sigma(v) - \mathtt{In}(v)|.$$

Lemma 2 *For any permutation* $\sigma : [n] \to [n]$,

$$\sum_{v\in[n]} |\sigma(v) - \mathtt{In}(v)| \geq \sum_{v\in[n]} |\mathcal{A}(v) - \mathtt{In}(v)|.$$

Lemma 3 *For any two permutations* $\sigma, \rho : [n] \to [n]$,

$$\sum_{v\in[n]} |\sigma(v) - \rho(v)| \geq |B_\rho - B_\sigma|.$$

We first show how the above lemmas prove our main result.

Proof of Theorem 1: Consider the following sequence of inequalities.

$$\begin{aligned}
4B_{\mathcal{O}} &\geq \sum_{v\in[n]} |\mathcal{O}(v) - \mathtt{In}(v)| + \sum_{v\in[n]} |\mathcal{O}(v) - \mathtt{In}(v)| \\
&\geq \sum_{v\in[n]} |\mathcal{O}(v) - \mathtt{In}(v)| + \sum_{v\in[n]} |\mathcal{A}(v) - \mathtt{In}(v)| \\
&= \sum_{v\in[n]} (|\mathcal{O}(v) - \mathtt{In}(v)| + |\mathcal{A}(v) - \mathtt{In}(v)|) \\
&\geq \sum_{v\in[n]} |\mathcal{O}(v) - \mathcal{A}(v)| \\
&\geq B_{\mathcal{A}} - B_{\mathcal{O}}
\end{aligned}$$

The first, second and last inequalities follow from Lemmas 1, 2 and 3 respectively (with $\sigma = \mathcal{O}$ and $\rho = \mathcal{A}$). The third inequality is triangle inequality while the equality just follows from rearrangement of the terms. Thus, we have $B_{\mathcal{A}} \leq 5B_{\mathcal{O}}$ which proves the theorem. ∎

In the rest of this section, we will prove Lemmas 1-3.

Proof of Lemma 1: Consider any arbitrary $v \in [n]$. Let $W_L^-(v)$ be the sum of weights of edges from vertices to the "left" of v (according to σ) to v; $W_L^+(v)$ be the sum of weights of edges from v to vertices which are to the left of v ; and $W_R^-(v)$ be the sum of weights of edges from vertices which are to the right of v to v. More formally,

$$W_L^-(v) = \sum_{u:\sigma(u)<\sigma(v)} w_{uv}$$

$$W_L^+(v) = \sum_{u:\sigma(u)<\sigma(v)} w_{vu}$$

$$W_R^-(v) = \sum_{u:\sigma(u)>\sigma(v)} w_{uv}$$

By definition, we have

$$W_L^-(v) + W_R^-(v) = \mathtt{In}(v) \tag{2}$$

The following identity follows from definitions and the fact that weights satisfy probability constraints.

$$W_L^+(v) + W_L^-(v) = \sigma(v) \tag{3}$$

Now, $2B_\sigma = \sum_{v\in[n]}(W_L^+(v) + W_R^-(v))$ as each backedge is counted twice in the sum. To complete the proof, we claim that for any $v \in [n]$, $W_L^+(v) + W_R^-(v) \geq |\sigma(v) - \mathtt{In}(v)|$.

Indeed from (2) and (3),

$$\begin{aligned}
W_L^+(v) + W_R^-(v) &= \sigma(v) + \mathtt{In}(v) - 2W_L^-(v) \\
&= |\sigma(v) - \mathtt{In}(v)| + 2(\min\{\sigma(v), \mathtt{In}(v)\} - W_L^-(v)) \\
&\geq |\sigma(v) - \mathtt{In}(v)|
\end{aligned}$$

The last inequality again follows from (2) and (3) and the fact that $W_L^+(v), W_R^-(v) \geq 0$. ■

Lemma 2 is a restatement of the fact that for any real numbers $a_1 \leq a_2 \cdots \leq a_n$, the permutation $\sigma : [n] \to [n]$ which minimizes the quantity $\sum_{i=1}^n |a_i - \sigma(i)|$ is the identity. For the sake of completeness, we present a proof in Appendix A.

Proof of Lemma 3: Consider the set of edges which are back edges in σ but are not backedges in ρ. Denote this set by $\mathcal{B}_{\sigma\setminus\rho}$. Also consider the set of edges which are back edges in ρ but are not backedges in σ. Denote this set by $\mathcal{B}_{\rho\setminus\sigma}$. Note that

$$\sum_{(u,v)\in\mathcal{B}_{\sigma\setminus\rho}} w_{uv} + \sum_{(u,v)\in\mathcal{B}_{\rho\setminus\sigma}} w_{uv} \geq |B_\rho - B_\sigma| \tag{4}$$

The crucial observation is that if an edge $(u,v) \in \mathcal{B}_{\sigma\setminus\rho}$ then $(v,u) \in \mathcal{B}_{\rho\setminus\sigma}$. This along with the fact that the weights satisfy probability constraints imply that $\mathcal{K}(\sigma,\rho) = \sum_{(u,v)\in\mathcal{B}_{\sigma\setminus\rho}} w_{uv} + \sum_{(u,v)\in\mathcal{B}_{\rho\setminus\sigma}} w_{uv}$ which by (4) implies $\mathcal{K}(\sigma,\rho) \geq |B_\rho - B_\sigma|$. Equation (1) completes the proof. ■

4 A Lower Bound for INCR-INDEG

We will prove the following theorem in this section.

Theorem 2 *For every constant $\epsilon > 0$, there exists an infinite family of (unweighted) tournaments $\mathcal{T}_\epsilon$ such that arranging the vertices of any tournament in $\mathcal{T}_\epsilon$ according to their indegrees,* irrespective *of how ties are broken, results in at least $5 - \epsilon$ times as many backedges as the optimal ordering.*

Note that the above result implies the analysis in Section 3 is tight, even if one modified INCR-INDEG to break ties in some "intelligent" way.

For any tournament T, we will use $\mathcal{I}(T)$ to denote the ordering according to indegrees which induces the least number of backedges (that is, ties are broken "optimally"). Also let $\mathcal{O}(T)$ denote the optimal ordering. For the rest of this section we use tournaments to refer to unweighted tournaments.

We will use two parameters, x and n, in this section. For any $n \geq 5$ and $x \geq 4$ such that x is a perfect square, we will construct a tournament $T_{x,n}$ such that

$$\lim_{x,n\to\infty} \frac{B_{\mathcal{I}(T_{x,n})}}{B_{\mathcal{O}(T_{x,n})}} = 5, \tag{5}$$

which will prove Theorem 2.

In the rest of this section, we will describe the construction of $T_{x,n}$ and show that (5) holds. Fix $n \geq 5$ and $x \geq 4$ such that x is a perfect square. $T_{x,n}$ will have $n(2x+1)$ vertices. We will partition the vertices into n *block*s of $2x+1$ vertices each. The i^{th} block for any $i = 1, 2, \ldots, n$, will be denoted by b^i. Further, for every $j = 0, 1, \ldots, 2x$; the j^{th} node in b^i will be denoted by b^i_j. The node b^i_x is the *middle node* of b^i and the sets of nodes $\{b^i_0, b^i_1, \cdots, b^i_{x-1}\}$ and $\{b^i_{x+1}, b^i_{x+2}, \cdots, b^i_{2x}\}$ are the *left half* and *right half* of b^i respectively. Let ϕ denote the ordering $b^1_0, \cdots, b^1_{2x}, b^2_0, \cdots, b^2_{2x}, \cdots, b^n_0, \cdots, b^n_{2x}$. Finally, unless mentioned otherwise, an edge will be called backward or forward in $T_{x,n}$ with respect to ϕ.

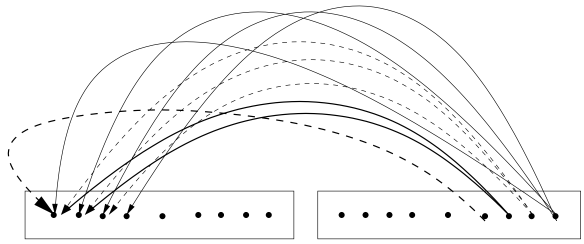

Figure 1: The `Type I` edges between b^i and b^{i+1} $(1 \le i < n)$ when $x = 4$.

The basic idea behind the construction is as follows. In $T_{x,n}$, the sub-tournaments spanned by each b^i would have no backedges if nodes in b^i are arranged according to ϕ. However, backedges from b^{i+1} and b^{i+2} will force $\mathcal{I}(T_{x,n})$ to order the vertices in b^i (more or less) in the reverse order of ϕ. This will result in $\mathcal{I}(T_{x,n})$ inducing many more back edges than the optimal.

We will describe the construction of $T_{x,n}$ by starting with a tournament on the vertex set $\cup_{i=1}^{n} b^i$ such that all the edges are forward edges according to ϕ. Then we will reverse the direction of some edges between b^i and b^{i+1} (which we call `Type I` edges) and some edges between b^i and b^{i+2} (which we call `Type II` edges) to get our final $T_{x,n}$. We now formally define these edges.

Assume we start with a set of edges E on $V = \cup_{i=1}^{n} b^i$ such that for any $u, v \in V$, $(u,v) \in E$ if $\phi(u) < \phi(v)$ and $(v,u) \in E$ otherwise. We first describe the `Type I` edges. For every i $(1 \le i < n)$, the last vertex of b^{i+1} has a `Type I` edge to every vertex in the left half of b^i. The second last vertex of b^{i+1} has a `Type I` edge to all but the last vertex in the left half of b^i and so on. More formally, for every $i = 1, 2, \cdots, n-1$,

`for` $(j = x+1, x+2, \cdots, 2x)$

`for` $(k = 0, 1, \cdots, j - x - 1)$

$$E \leftarrow (E \setminus \{(b^i_k, b^{i+1}_j\}) \cup \{(b^{i+1}_j, b^i_k)\}$$

See Figure 1 for an example when $x = 4$.

We turn to the `Type II` edges. First, partition the left and the right half of every b^i into $\sqrt{x}$ consecutive *minigroups* of $\sqrt{x}$ vertices each. A minigroup is *connected* to another if there is an edge from the ℓth $(0 \le \ell \le \sqrt{x} - 1)$ vertex in the first minigroup to the ℓth vertex in the second minigroup. For any i $(1 \le i < n-1)$, `Type II` edges are introduced to connect the last minigroup in the right half of b^{i+2} to all the minigroups in the left half of b^i. The second last minigroup in the right half of b^{i+2} is connected to all but the last minigroup in the left half of b^i and so on. More formally. for every $i = 1, 2, \cdots, n-2$,

`for` $(k = 0, 1, \cdots, \sqrt{x} - 1)$

`for` $(r = 0, 1, \cdots, k)$

`for` $(\ell = 0, 1 \cdots, \sqrt{x} - 1)$

$$E \leftarrow (E \setminus \{(b^i_{r\sqrt{x}+\ell}, b^{i+2}_{x+k\sqrt{x}+\ell+1})\}) \cup \{(b^{i+2}_{x+k\sqrt{x}+\ell+1}, b^i_{r\sqrt{x}+\ell})\}$$

See Figure 2 for an example of `Type II` edges for the case when $x = 4$.

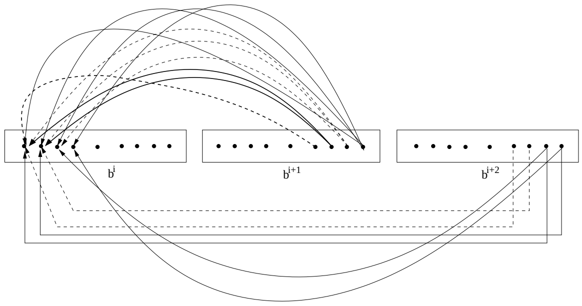

Figure 2: `Type I` edges between b^i and b^{i+1} and `Type II` edges between b^i and b^{i+2} ($1 \leq i < n-1$) when $x = 4$. `Type II` edges are the ones which are not present in Figure 1.

The tournament defined by the vertices V and edges E is the required tournament $T_{x,n}$. We will now estimate the indegrees of the vertices in $T_{x,n}$. Consider an i such that $2 < i < n-2$. Before `Type I` and `Type II` edges were introduced, $T_{x,n}$ was an acyclic graph. Thus, the indegree of the vertex b^i_j (where $0 \leq j \leq 2x$) was the number of vertices connected to it, that is, $b^i_j = (i-1)(2x+1)+j$. When `Type I` edges were introduced, the indegree of the last vertex in b^i decreased by x (as there was now an edge from it to every vertex in the left half of b^{i-1}) while the indegree of the first vertex in b^i increased by x (as there was now an edge from every vertex in the right half of b^{i+1} to b^i_0). Similarly, the indegree of the second last vertex decreased by $x-1$ while the indegree of the second vertex increased by $x-1$ and so on. Thus, after all `Type I` edges were introduced, the degree of b^i_j ($0 \leq j \leq 2x$) was $(i-1)(2x+1)+x$. When `Type II` edges were introduced, the indegree of every vertex in the last minigroup in the right half of b^i decreased by $\sqrt{x}$ (as the last minigroup was now connected to every minigroup in the left half of b^{i-2}) while the indegree of every vertex in the first minigroup in the left half of b^i increased by $\sqrt{x}$ (as every minigroup in the right half of b^{i+2} was now connected to the first minigroup in the left half of b^i). Similarly, the indegree of every vertex in the second last minigroup in the right half of b^i decreased by $\sqrt{x}-1$ while the indegree of every vertex in the second minigroup in the left half of b^i increased by $\sqrt{x}-1$ and so on. In particular, the indegree of b^i_x did not change. Thus, for $0 \leq j < x$, $b^i_j = (i-1)(2x+1)+x+\lceil \frac{x-j}{\sqrt{x}} \rceil$ and for $x < j \leq 2x$, $b^i_j = (i-1)(2x+1)+x-\lceil \frac{j-x}{\sqrt{x}} \rceil = (i-1)(2x+1)+x+\lfloor \frac{x-j}{\sqrt{x}} \rfloor$.

Taking care of the boundary cases we have the following expressions for the indegrees.

$$\texttt{In}(b^1_j) = \begin{cases} x + \lceil \frac{x-j}{\sqrt{x}} \rceil, & j \in [0,x] \\ j, & j \in (x, 2x] \end{cases} \tag{6}$$

$$\texttt{In}(b^2_j) = \begin{cases} 3x+1+\lceil \frac{x-j}{\sqrt{x}} \rceil, & j \in [0,x] \\ 3x+1, & j \in (x, 2x] \end{cases} \tag{7}$$

For $i = 3, \cdots, n-2$ and for $j = 0, 1, \cdots, 2x$;

$$\texttt{In}(b^i_j) = \begin{cases} (i-1)(2x+1)+x+\lceil \frac{x-j}{\sqrt{x}} \rceil, & j \in [0,x] \\ (i-1)(2x+1)+x+\lfloor \frac{x-j}{\sqrt{x}} \rfloor, & j \in (x, 2x] \end{cases} \tag{8}$$

$$\texttt{In}(b^{n-1}_j) = \begin{cases} (n-2)(2x+1)+x, & j \in [0,x] \\ (n-2)(2x+1)+x+\lfloor \frac{x-j}{\sqrt{x}} \rfloor, & j \in (x, 2x] \end{cases} \tag{9}$$

$$\mathtt{In}(b_j^n) = \begin{cases} (n-1)(2x+1)+j, & j \in [0,x] \\ (n-1)(2x+1)+x+\lfloor \frac{x-j}{\sqrt{x}} \rfloor, & j \in (x,2x] \end{cases} \tag{10}$$

We first upper bound the number of backedges in the optimal ordering.

Lemma 4 *The number of backedges in the optimal ordering of $T_{x,n}$ is at most $\frac{x^2n}{2}+o(x^2n)$.*

Proof: To prove the lemma, we show that $B_\phi \leq x^2n/2+o(x^2n)$. Note that the only backedges in $T_{x,n}$ according to ϕ are the `Type I` and `Type II` edges. By definition, the number of `Type I` edges is

$$(n-1)(x+x-1+\cdots+1) = \frac{x(x-1)}{2}(n-1) \leq \frac{x^2n}{2}$$

and the number of `Type II` edges is

$$(n-2)(\sqrt{x}+\sqrt{x}-1+\cdots+1) = \frac{x(\sqrt{x}-1)}{2}(n-2) \leq \frac{x^{3/2}n}{2} = o(x^2n).$$

The proof is complete. ■

We now lower bound the number of backedges induced by $\mathcal{I}(T_{x,n})$.

Lemma 5 *The number of backedges induced by $\mathcal{I}(T_{x,n})$ on $T_{x,n}$ is at least $\frac{5x^2n}{2}-o(x^2n)$.*

Note that Lemmas 4 and 5 prove equation (5) and thus, Theorem 2. We end this section by proving Lemma 5.

Proof of Lemma 5: We first claim that for any $i < i'$, every node in b^i is placed before every node of $b^{i'}$ by $\mathcal{I}(T_{x,n})$. In other words, all `Type I` and `Type II` edges are back edges (between vertices in the different blocks) according to $\mathcal{I}(T_{x,y})$. The proof of lemma 4 shows that this number is at least $x^2n/2-o(x^2n)$. For any i, let max_i and min_i be the maximum and minimum indegrees of all vertices in b^i. To prove the claim, we will show that

$$\text{for all } i=1,2,\cdots,n-1; \quad max_i < min_{i+1}. \tag{11}$$

Indeed from Equations (6)-(10), we have the following values for max_i and min_i:

$$max_i = \begin{cases} 2x, & i=1 \\ (2i-1)x+\sqrt{x}+(i-1), & i=2,\cdots,n-2 \\ (2n-3)x+(n-2), & i=n-1 \end{cases}$$

$$min_i = \begin{cases} 3x+1, & i=2 \\ (2i-1)x-\sqrt{x}+(i-1), & i=3,\cdots,n \\ (2n-2)x+n-1, & i=n \end{cases}$$

An inspection of the values shows that (11) holds.

Thus, we have counted all the back edges between vertices of b^i and $b^{i'}$ for $i \neq i'$. We now need to count the number of back edges between vertices in the same b^i. Counting conservatively, we assume that there are no such back edges for $i \in \{1,2,n,n-1\}$. Fix an i such that $2 < i < n-1$. We claim that the number of back edges between vertices in b^i is at least

$$x(2x+1)-x(\sqrt{x}-1). \tag{12}$$

To see this divide the left half of b^i into $\sqrt{x}$ minigroups– $l_0, l_1, \cdots, l_{\sqrt{x}-1}$. In particular, l_k consists of the vertices $b^i_{k\sqrt{x}}, b^i_{k\sqrt{x}+1}, \cdots, b^i_{(k+1)\sqrt{x}-1}$. Similarly the right half of b^i is divided into $\sqrt{x}$ minigroups– $r_0, r_1, \cdots, r_{\sqrt{x}-1}$. Observe from (8) that for any $k = 0, 1, \cdots, \sqrt{x}-1$; the degree of a vertex in l_k and r_k is $(i-1)(2x+1) + x + \sqrt{x} - k$ and $(i-1)(2x+1) + x - k - 1$ respectively. Thus, $\mathcal{I}(T_{x,n})$ will have to arrange vertices in the order $r_{\sqrt{x}-1}, r_{\sqrt{x}-2}, \cdots, r_0$, followed by the middle node b^i_x, followed by vertices in the order $l_{\sqrt{x}-1}, l_{\sqrt{x}-2}, \cdots, l_0$. Again counting conservatively, we assume that there are no backedges in induced tournaments over any minigroup l_k or r_k (where $k = 0, 1, \cdots, \sqrt{x}-1$). However, note that every other edge in $T^i_{x,n}$, the induced tournament over b^i, is a backedge. There are a total of $\binom{2x+1}{2} = x(2x+1)$ edges in $T^i_{x,n}$ while the induced tournaments over any l_k or r_k has $\binom{\sqrt{x}}{2}$ many edges and there are $2\sqrt{x}$ such minigroups. This implies that the number of backedges in $T^i_{x,n}$ is at least $x(2x+1) - 2\sqrt{x} \cdot \sqrt{x}(\sqrt{x}-1)/2$ as claimed in (12).

Recalling that there are $n-4$ choices for i, the number of backedges within some block totaled over all the $n-4$ blocks is

$$\big(x(2x+1) - x(\sqrt{x}-1)\big)\,(n-4) \geq 2x^2 n - o(x^2 n).$$

Adding the estimates of the number of backedges between different b^is and number of backedges within the same b^i completes the proof. ∎

5 Conclusions and Open problems

The best known approximation guarantee for the FAS-TOURNAMENT problem is 3 for deterministic algorithms ([18]) and 2.5 for randomized algorithms ([1]). It is an interesting open question to determine the correct approximation factor. We remark that the complementary problem of maximum acyclic subgraph problem on tournaments has a PTAS [3].

Another interesting problem is to get a tighter bound on how good Borda's method is for approximating the optimal ranking for rank aggregation. Note that the example in Section 4 is not a valid exmaple for rank aggregation (the weights do not satisfy triangle inequality).

Acknowledgments

We would like to thank Moses Charikar and Venkatesan Guruswami for helpful discussions and Vincent Conitzer for pointing out Borda's method to us.

References

[1] N. Ailon, M. Charikar, and A. Newman. Aggregating inconsistent information: Ranking and clustering. In *Proceedings of the 37th ACM Symposium on Theory of Computing (STOC)*, pages 684–693, 2005.

[2] N. Alon. Ranking tournaments. In *SIAM Journal on Discrete Math*, To Appear.

[3] S. Arora, A. Frieze, and H. Kaplan. A new rounding procedure for the assignment problem with applications to dense graph arrangement problems. In *Proceedings of the 37th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 24–33, 1996.

[4] J. Bang-Jensen and C. Thomassen. A polynomial algorithm for 2-path problem in semicomplete graphs. *SIAM Journal of Discrete Math*, 5(3):366–376, 1992.

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

[6] B. Berger and P. W. Shor. Approximation algorithms for the maximum acyclic subgraph problem. In *Proceedings of the 1st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 236–243, 1990.

[7] J. C. Borda. Mémoire sur les élections au scrutin. *Histoire de l'Académie Royale des Sciences*, 1781.

[8] V. Conitzer. Computing Slater Rankings Using Similarities Among Candidates. *IBM Research Report No. RC23748*, 2005.

[9] P. Diaconis and R. Graham. Spearman's footrule as a measure of disarray. *Journal of the Royal Statistical Society, Series B*, 39(2):262–268, 1977.

[10] I. Dinur and S. Safra. On the importance of being biased. In *Proceedings of the 34th ACM Symposium on Theory of Computing (STOC)*, pages 33–42, 2002.

[11] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar. Rank aggregation methods for the web. In *Proceedings of the 10th International Conference on the World Wide Web (WWW10)*, pages 613–622, 2001.

[12] G. Even, J. S. Naor, M. Sudan, and B. Schieber. Approximating minimum feedback sets and multicuts in directed graphs. *Algorithmica*, 20(2):151–174, 1998.

[13] R. Fagin, R. Kumar, M. Mahdian, D. Sivakumar, and E. Vee. Rank Aggregation: An Algorithmic Perspective. Unpublished Manuscript, 2005.

[14] R. Hassin and S. Rubinstein. Approximations for the maximum acyclic subgraph problem. *Information Processing Letters*, 51:133–140, 1977.

[15] R. Karp. Reducibility among combinatorial problems. *Complexity of Computer Computations*, pages 85–104, 1972.

[16] C. Papadimitriou and M. Yannakakis. Optimization, approximation, and complexity classes. *Journal of Computer System Science*, 43:425–440, 1991.

[17] P. Seymour. Packing directed circuits fractionally. *Combinatorica*, 15(2):281–288, 1995.

[18] A. van Zuylen. Deterministic approximation algorithms for ranking and clusterings. *Cornell ORIE Tech Report No. 1431*, 2005.

A Proof of Lemma 2

If σ sorts the vertices in $[n]$ according to their indegrees then the statement of the lemma holds trivially.

So now consider the case when there exits $i \in [n]$ such that $\texttt{In}(u) > \texttt{In}(v)$ where $u = \sigma^{-1}(i)$ and $v = \sigma^{-1}(i+1)$. Construct a new ordering σ' that is same as σ except u and v are swapped: $\sigma'(w) = \sigma(w)$ if $w \notin \{u, v\}$ and $\sigma'(u) = i+1$, $\sigma'(v) = i$. We next show that $\sum_{v\in[n]} |\sigma(v) - \texttt{In}(v)| \geq \sum_{v\in[n]} |\sigma'(v) - \texttt{In}(v)|$: the rest of the proof is a simple induction. By the construction of σ',

$$\begin{aligned}
&\sum_{v\in[n]} (|\sigma(v) - \texttt{In}(v)| - |\sigma'(v) - \texttt{In}(v)|) \\
&\quad = |i - \texttt{In}(u)| + |i + 1 - \texttt{In}(v)| - |i - \texttt{In}(v)| - |i + 1 - \texttt{In}(u)| \\
&\quad = 2(\min\{i, \texttt{In}(v)\} - \min\{i, \texttt{In}(u)\}) + 2(\min\{i + 1, \texttt{In}(u)\} - \min\{i + 1, \texttt{In}(v)\})
\end{aligned}$$

The last equality follows from the identity $|x - y| = x + y - 2\min\{x, y\}$. Finally it can be verified[4] that the last sum is always non-negative. ■

[4] There are three cases. If $i \geq \texttt{In}(u)$ then the first term is $2(\texttt{In}(v) - \texttt{In}(u))$ while the second term is $2(\texttt{In}(u) - \texttt{In}(v))$. If $\texttt{In}(v) \geq i$ then the first term is 0 while the second term is $2\max(i + 1 - \texttt{In}(v), 0)$. Finally if $\texttt{In}(u) > i > \texttt{In}(v)$ then the first term is $2(\texttt{In}(v) - i)$ while the second term is $2(i + 1 - \texttt{In}(v))$.

ECCC ISSN 1433-8092
http://www.eccc.uni-trier.de/eccc
ftp://ftp.eccc.uni-trier.de/pub/eccc
ftpmail@ftp.eccc.uni-trier.de, subject 'help eccc'