 Cynthia Day, sunet: cyndia

Ordering by weighted number of wins gives a good ranking for weighted tournaments *[Coppersmith/Fleischer/Rurda]*

This paper presents an algorithm for ordering the nodes in a weighted tournament and analyzes its performance guarantees. We begin by giving a little background:

The idea of a tournament is very intuitive. The nodes can be thought of as representing teams, and the edges between them encodes their performance against one another. For example, given n teams where every pair of team plays a single game, you get an unweighted, directed graph where there is an edge pointing from team B to team A if A won against B.

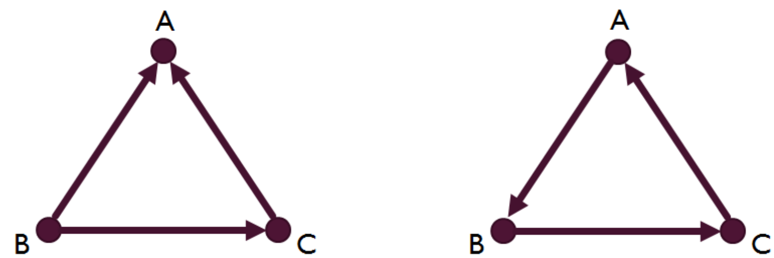

Fig 1. Examples of unweighted tournaments with three teams

In Figure 1, we have unweighted tournaments with three teams A, B, and C. In the example on the left, it is clear that A is the best team and B is the worst. However, if we have a three-cycle as on the left, the ordering becomes less clear.

Weighted tournaments derive from the same idea. Say teams A and B play multiple games against each other, of which A wins 1 and B wins 2. Then we can estimate the probability of A winning against B as 1/3 and the probability of B winning against A as 2/3. Weighted tournaments are complete directed graphs, where the edge from A to B represents the probability that B wins against A and vice versa. Thus, the two edges going between any pair of nodes should sum to 1. Note that the unweighted tournament is a special case of this where edge weights are either 1 or 0 and edges of weight 0 are left out.

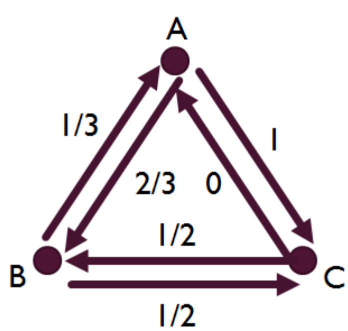

Fig 2. Example of a weighted tournament with three teams

Given a weighted tournament, a natural impulse is to determine the absolute ranking of the teams. This query is formalized through the minimum feedback arc set problem, which given a graph G(V,E) seeks to find the arc set E' with minimum weight such that (V, E\E') is acyclic. This is equivalent to finding an ranking σ of the vertices, or a mapping from V to {0, 1, … , n-1}, for which the weight of the back edges is minimized. A back edge is an edge pointing from a higher-ranked node in σ to a lower-ranked node. If we again view the nodes as teams, then a back edge is the probability that a higher-ranked node loses to a lower-ranked node. Since we want to come up with an accurate ranking of the teams, it makes sense that we want to minimize the probability of such instances. Lastly, when applied to tournaments, the minimum feedback arc set is abbreviated as FAS-tournament.

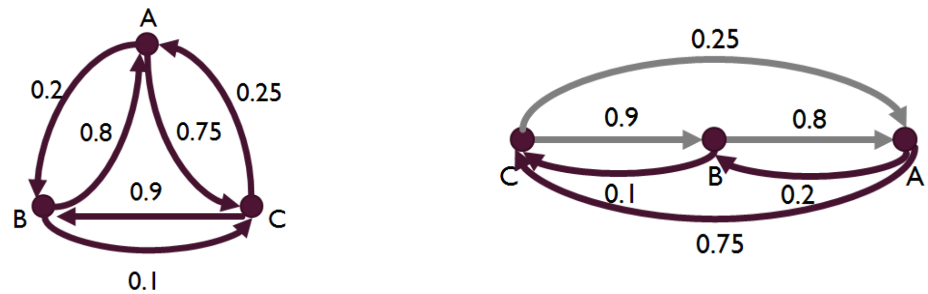

Fig 3. A weighted tournament and a rearrangement that reflects its optimal ordering

For example, in Figure 3 the graph on the left is a sample weighted tournament with three teams. Its optimal ordering happens to be σ(C) = 0, σ(B) = 1, and σ(A) = 2, and the sum of the back edges is $0.1 + 0.2 + 0.75 = 1.05$.

A large body of work has already been done on FAS-tournament. It was shown in 2006 to be NP-hard by separate groups of researchers. Slightly earlier in 2005, Ailon, Charikar, and Newman found a 3-approximation algorithm for unweighted FAS-tournament that performs recursion using pivot vertices and which gives a 5-approximation for the weighted case. Van Zuylen has provided a deterministic algorithm that is a 3-approximation for weighted FAS-tournament, and later collaborated with other researchers to provide a combinatorial deterministic algorithm that provides a 4-approximation. A polynomial-time approximation scheme has also been introduced by Chudnovsky, Seymour, and Sullivan for weighted tournaments; it can be used with loosened probability constraints (the edges between a pair of nodes can sum to less than 1, with running time doubly exponential in the reciprocal of the lower bound).

The algorithm explored in this paper is that of ranking nodes by their weighted indegree and is thus known as INCR-INDEG. An example is pictured below. This is a very natural approach because the indegree of a node is representative of how well it performs against other nodes. The authors of the paper demonstrate that this algorithm gives a 5-approximation on both weighted and unweighted FAS-tournament. While this is a worse performance guarantee than some of the past results, the simplicity of INCR-INDEG gives it an edge over the other algorithms.

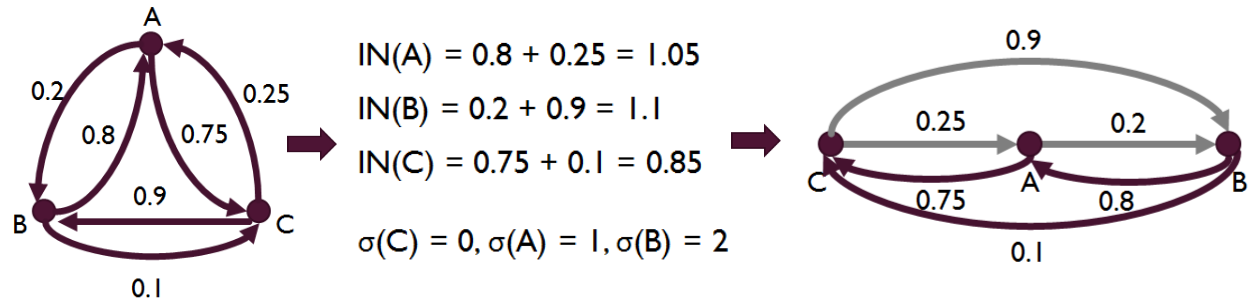

Fig 4. An example if INCR-INDEG applied to a small weighted tournament

Before going into the technical details of the analysis, we first introduce some notation. We let [n] stand for the set {0, 1, … , n-1}, that is, the vertex labels for a graph with n nodes. The weight of an edge going from vertex u to vertex v is denoted w_{uv}. The weighted indegree at a vertex v is represented as

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

So for example, for vertex A above, IN(A) = 1.05. In addition, σ:[n] → [n] represents some ordering of the vertex set of a graph. We let OPT stand for the optimal ordering, which minimizes the weight of the back edges, and we let INCR stand for the ordering output by INCR-INDEG. For a permutation σ, the weight of the back edges induced by σ is

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

Lastly, we define two different distance metrics for comparing permutations. Spearman's footrule distance is

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

and the Kendall-Tau distance is

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

Note that this latter expression is simply the number of ordered pairs of vertices ordered differently between the two permutations.

We move on to the proof that INCR-INDEG gives a 5-approximation. The paper splits this proof into three key lemmas:

$$2B_\sigma \geq \sum_v |\sigma(v) - IN(v)| \qquad (1)$$

$$\sum_v |\sigma(v) - IN(v)| \geq \sum_v |INCR(v) - IN(v)| \qquad (2)$$

$$\sum_v |\sigma(v) - \rho(v)| \geq |B_\rho - B_\sigma| \qquad (3)$$

First we demonstrate how these lemmas together show that $B_{INCR} \leq 5B_{OPT}$. From lemma (1), we have

$$4B_{OPT} \geq \sum_v |OPT(v) - IN(v)| + \sum_v |OPT(v) - IN(v)|$$

Substituting in lemma (2) gives

$$4B_{OPT} \geq \sum_v |OPT(v) - IN(v)| + \sum_v |INCR(v) - IN(v)|$$
$$= \sum_v (|OPT(v) - IN(v)| + |INCR(v) - IN(v)|)$$

Using the triangle inequality, we get that

$$4B_{OPT} \geq \sum_v |OPT(v) - INCR(v)| = F(OPT, INCR)$$

And finally, substituting in lemma (3) gives

$$4B_{OPT} \geq |B_{INCR} - B_{OPT}| \geq B_{INCR} - B_{OPT}$$

Which rearranges to

$$B_{INCR} \leq 5B_{OPT}$$

as desired.

Now we discuss the proofs of the three lemmas.

Lemma 1: $2B_\sigma \geq \sum_v |\sigma(v) - IN(v)|$

The proof for this lemma hinges upon a clever classification of the edges at each node in an ordering. Given a vertex v and a permutation σ, we refer to nodes ranked lower than v as being to the left of v and vertices ranked higher as being to the right of v. Thus, for example, back edges are all the edges pointing to the left. Then we let $W_L^-(v)$ be the sum of the weights of edges pointing towards v from the left, while $W_L^+(v)$ is the sum of the weights of the edges pointing away from v towards the left. Similarly, $W_R^-(v)$ is the sum of the weights of edges pointing towards v from the right, while $W_R^+(v)$ is the sum of the weights of the edges pointing away from v towards the right.

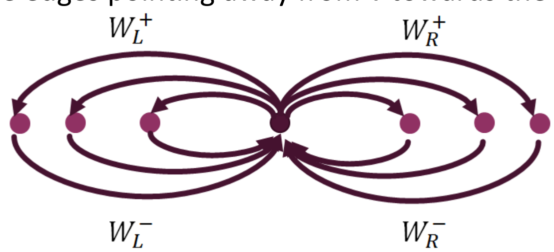

Fig 5. The different classifications of edges at a node

It follows that each of the expressions in lemma (1) can be expressed in terms of these new terms. First, the indegree of v is the sum of all the incoming edges, so

$$W_L^-(v) + W_R^-(v) = IN(v).$$

Next, the σ(v) is the rank of node v in our ordering, or the number of nodes to the left of v. Due to our probability constraints, the sum of the edges between v and any vertex u to the left of v is exactly 1. Thus,

$$W_L^-(v) + W_L^+(v) = \sigma(v).$$

Lastly, a back edge is any edge that points towards the left. Thus, the sum of all back edges that involve v is $W_L^+(v) + W_R^-(v)$. If we sum over all vertices, then we will have double-counted each back edge, giving us

$$2B_\sigma = \sum_{v\in[n]}\big(W_L^+(v) + W_R^-(v)\big).$$

Now, combining the expressions for indegree and rank gives us

$$W_L^+(v) + W_R^-(v) = \sigma(v) + IN(v) - 2W_L^-(v)$$

Substituting in $\sigma(v) + IN(v) = |\sigma(v) - IN(v)| + 2\min\{\sigma(v), IN(v)\}$, we get

$$W_L^+(v) + W_R^-(v) = |\sigma(v) - IN(v)| + 2\big(\min\{\sigma(v), IN(v)\} - W_L^-(v)\big)$$

Since $\min\{\sigma(v), IN(v)\} \geq W_L^-(v)$, it follows that

$$W_L^+(v) + W_R^-(v) \geq |\sigma(v) - IN(v)|.$$

Summing over all vertices v then gives

$$2B_\sigma = \sum_{v\in[n]}\big(W_L^+(v) + W_R^-(v)\big) \geq \sum_{v\in[n]}|\sigma(v) - IN(v)|$$

as desired.

Lemma 2: $\sum_v |\sigma(v) - IN(v)| \geq \sum_v |INCR(v) - IN(v)|$
First, note that this lemma is basically saying that to minimize pairwise differences between rank and indegree, the highest rank should be paired with the highest indegree and so forth. We demonstrate that for any permutation σ with vertices ordered differently, we can swap a pair of mismatching vertices and get a sum less than or equal to the previous sum. It follows inductively that INCR is the permutation that minimizes $\sum_v |\sigma(v) - IN(v)|$.

Assume σ is not equivalent to INCR. Then if we consider each pair of consecutive positions that σ maps to, there must be a pair whose indegrees are ordered oppositely. That is, there must exist σ(u) = i and σ(v) = i + 1 such that IN(u) > IN(v). If we consider a new ordering σ' that is identical to σ except that σ'(v) = i and σ'(u) = i + 1, then

$$\sum_v |\sigma(v) - IN(v)| - |\sigma'(v) - IN(v)|$$

sees the majority of its terms cancel out, leaving

$$= |i - IN(u)| + |i + 1 - IN(v)| - |i - IN(v)| - |i + 1 - IN(u)|$$

We can apply the identity $|x - y| = x + y - 2\min\{x, y\}$ to get

$$= i + IN(u) - 2\min\{i, IN(u)\} + i + 1 + IN(v) - 2\min\{i + 1, IN(v)\} - i - IN(v) + 2\min\{i, IN(v)\} - i - 1 - IN(u) + 2\min\{i + 1, IN(u)\}$$

$$= 2\,(\min\{i, IN(v)\} - \min\{i, IN(u)\}) + 2(\min\{i + 1, IN(u)\} - \min\{i + 1, IN(v)\})$$

We can then perform some casework:

If $i \geq IN(u) > IN(v)$, then the above expression would become $2\big(IN(v) - IN(u)\big) + 2\big(IN(u) - IN(v)\big) = 0$

If $IN(u) > IN(v) \geq i$, then the first term becomes 0 and the second term is guaranteed to be nonnegative since IN(u) > IN(v). Thus, the entire expression must be nonnegative.

If $IN(u) > i > IN(v)$, then the first term becomes $2(IN(v) - i)$ and the second term becomes either $2\big(IN(u) - IN(v)\big)$ or $2\big(i + 1 - IN(v)\big)$. Either way, the value of the second term is at least $2(i - IN(v))$, thus the expression is also nonnegative in this case.

It follows that

$$\sum_v |\sigma(v) - IN(v)| - |\sigma'(v) - IN(v)| \geq 0$$

as desired.

Lemma 3: $\sum_v |\sigma(v) - \rho(v)| \geq |B_\rho - B_\sigma|$
We let $B_{\sigma\setminus\rho}$ denote the set of back edges induced by permutation σ but not by permutation ρ. Similarly, $B_{\rho\setminus\sigma}$ denotes the set of back edges in ρ but not in σ. Then

$$|B_\rho - B_\sigma| = \left|\sum_{(u,v)\in B_{\sigma\setminus\rho}} w_{uv} - \sum_{(u,v)\in B_{\rho\setminus\sigma}} w_{uv}\right|,$$

so

$$\sum_{(u,v)\in B_{\sigma\setminus\rho}} w_{uv} + \sum_{(u,v)\in B_{\rho\setminus\sigma}} w_{uv} \geq |B_\rho - B_\sigma|.$$

Next, note that if $(u, v) \in B_{\sigma\setminus\rho}$, then by definition (u, v) does not belong to B_ρ, so it must be a forward edge in ρ. That means that (v, u) is a back edge in ρ, and thus $(v, u) \in B_{\rho\setminus\sigma}$. Furthermore, notice that a back edge will not be shared between the two permutations if and only if the vertices involved are ordered differently between the two permutations. So any pair of such vertices u and v contribute a sum of $w_{uv} + w_{vu} = 1$ to the sum of weights over $B_{\sigma\backslash\rho}$ and $B_{\rho\backslash\sigma}$. It follows that

$$K(\sigma, \rho) = \sum_{(u,v)\in B_{\sigma\backslash\rho}} w_{uv} + \sum_{(u,v)\in B_{\rho\backslash\sigma}} w_{uv} \geq |B_\rho - B_\sigma|.$$

At this point, we can use a theorem proved by Diaconis and Graham that $K(\sigma, \rho) \leq F(\sigma, \rho)$, thus

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

as desired.

This concludes the proof that INCR-INDEG provides a 5-approximation for the FAS-tournament problem.

Next, it is interesting to note that INCR-INDEG also provides a 5-approximation for RANK-AGGREGATION. RANK-AGGREGATION takes as input a list of permutations $\{\pi_1, \pi_2, \cdots, \pi_k\}$ and outputs the permutation σ for which $\sum_{i=1}^{k} K(\sigma, \pi_i)$ is minimized (recall that K is the Kendall Tau distance, or the number of pairs of nodes that are ordered differently between the two permutations). RANK-AGGREGATION can be reduced to weighted FAS-tournament in the following way: if the permutations given are on the node set [n], then we construct a weighted tournament on [n] with edges weighted such that w_{ij} is the fraction of the permutations that rank j higher than i. The INCR-INDEG method can then be used on this construction. Note that the number of ordering mismatches is proportional to the weight of the back edges (each back edge represents the fraction of permutations that disagree with a certain ordering), so the 5-approximation translates over to the results for RANK-AGGREGATION.

The final section of the paper is devoted to proving that 5-approximation is a tight bound. As this section was not touched upon in the presentation, we cover it in slightly less detail here.

The formal statement of the result is that for every positive constant ε, there are infinite families of unweighted tournaments such that ordering the vertices by indegree results in a factor of $5 - \epsilon$ more back edges. The authors prove this using a construction that for an integer $n \geq 5$ and perfect square $x \geq 4$ and creates a tournament $T_{x,n}$ such that

$$\lim_{x,n\to\infty} \frac{B_{INCR(T_{x,n})}}{B_{OPT(T_{x,n})}} = 5.$$

Let $T_{x,n}$ have $n(2x + 1)$ vertices. These vertices are then split into n equally sized blocks, where the i^{th} block is denoted b^i and the j^{th} node in b^i is denoted b_j^i. Then b_x^i is the middle node of b^i, with the lower-indexed nodes constituting the left half and the higher-indexed nodes constituting the right half. Lastly, let σ be the ordering $b_0^1, \cdots, b_{2x}^1, b_0^2, \cdots, b_{2x}^2, \cdots, b_0^n, \cdots, b_{2x}^n$. We describe edges as forwards/backwards with respect to σ.

Next, we begin our construction with an unweighted tournament $T_{x,n}$ such that all edges are forward edges. We then convert some of these edges to Type I edges by drawing edges from b_{2x}^{i+1} to all x nodes in the left half of b^i, from b_{2x-1}^{i+1} to the first $x - 1$ nodes in b^i, b_{2x-2}^{i+1} to the first $x - 2$ nodes in b^i, and so on.

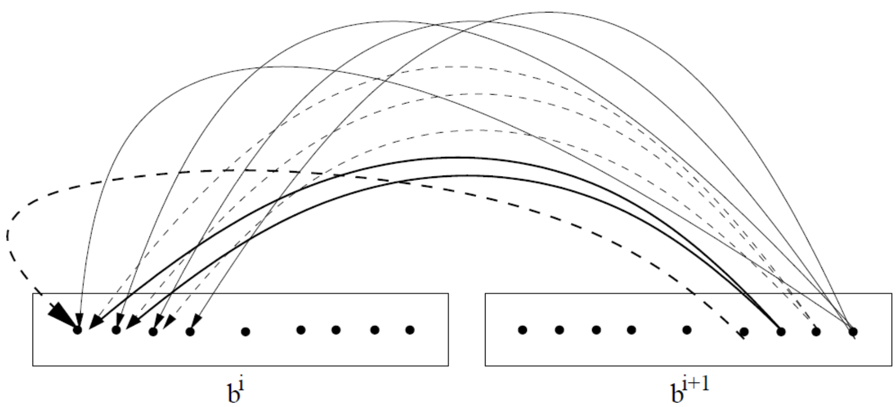

Fig 6. Type I edges between b^i and b^{i+1}

We also construct Type II edges on our tournament. First we divide the right and left halves of the blocks into $\sqrt{x}$ minigroups, each with $\sqrt{x}$ vertices. We say that two minigroups are connected if for all j between 0 and $\sqrt{x}-1$, there is an edge from the j^{th} vertex of the first minigroup to the j^{th} vertex of the other. We add Type II edges such that the last minigroup of the right half of b^{i+2} is connected to all $\sqrt{x}$ minigroups in the left half of b^i, the second to last minigroup of the right half of b^{i+2} is connected to the first $\sqrt{x}-1$ minigroups in the left half of b^i, the third to last minigroup of the right half of b^{i+2} is connected to the first $\sqrt{x}-2$ minigroups in the left half of b^i, and so on.

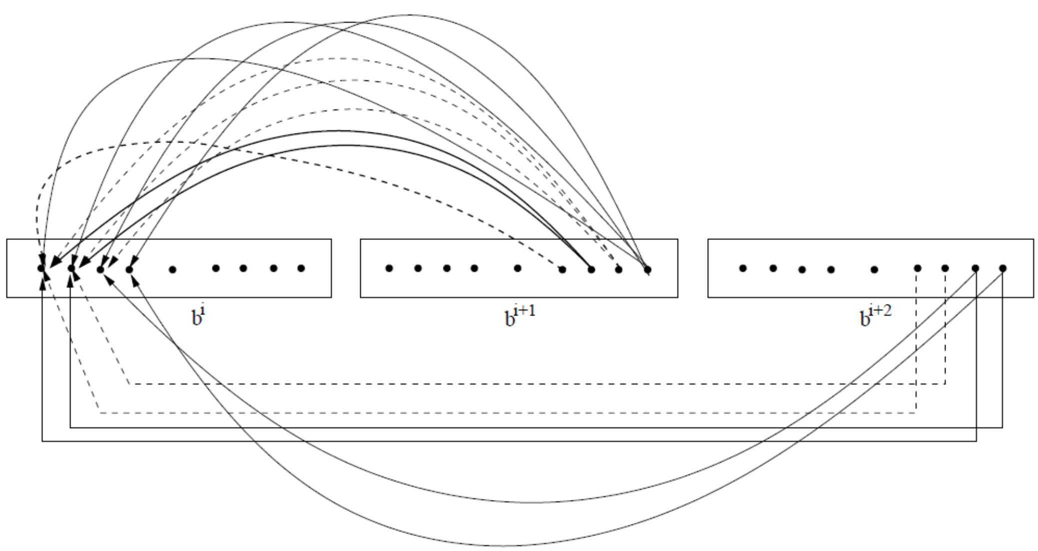

Fig 7. Type I and Type II edges between b^i, b^{i+1}, and b^{i+2}.

We can now calculate the indegree of each vertex. Note that before Type I and Type II edges were added, each vertex b_j^i had indegree of $(i-1)(2x+1)+j$ (or exactly its ranking in the ordering σ). With the addition of Type I edges, the indegree of the last vertex in a block decreased by x, while the indegree of the first vertex in the previous block increased by x; the indegree of the second to last vertex in a block decreased by x-1, while the indegree of the second vertex in the previous block increased by x-1, and so on. Nodes in the first block do not experience a decrease in indegree, and nodes in the last block do not experience an increase in indegree. Similarly, with the addition of Type II edges, the indegrees of the vertices in the last minigroup in the right half of a block are decreased by $\sqrt{x}$ while indegrees of vertices in the first minigroup in the left half of the second block over are increased by $\sqrt{x}$; the indegrees of the vertices in the second to last minigroup in the right half of a block are decreased by $\sqrt{x}-1$ while indegrees of vertices in the second minigroup in the left half of the second block over are increased by $\sqrt{x}-1$, and so on. Once again, nodes in the edge blocks experience slightly different changes, ultimately leading to the following indegrees:

$$IN(b_j^1)\begin{cases} x+\left\lceil \frac{x-j}{\sqrt{x}} \right\rceil, j \in [0,x] \\ j, j \in (x,2x] \end{cases}$$

$$IN(b_j^2)\begin{cases} 3x+1+\left\lceil \frac{x-j}{\sqrt{x}} \right\rceil, j \in [0,x] \\ 3x+1, j \in (x,2x] \end{cases}$$

For $i \in [3, n-2]$,

$$IN(b_j^i)\begin{cases} (i-1)(2x+1)+x+\left\lceil \frac{x-j}{\sqrt{x}} \right\rceil, j \in [0,x] \\ (i-1)(2x+1)+x+\left\lfloor \frac{x-j}{\sqrt{x}} \right\rfloor, j \in (x,2x] \end{cases}$$

$$IN(b_j^{n-1})\begin{cases} (n-2)(2x+1)+x, j \in [0,x] \\ (n-2)(2x+1)+x+\left\lfloor \frac{x-j}{\sqrt{x}} \right\rfloor, j \in (x,2x] \end{cases}$$

$$IN(b_j^n)\begin{cases} (n-1)(2x+1)+j, j \in [0,x] \\ (n-1)(2x+1)+x+\left\lfloor \frac{x-j}{\sqrt{x}} \right\rfloor, j \in (x,2x] \end{cases}$$

Using these indegrees, we can bound the number of back edges in both the optimal solution and in the solution generated by INCR-INDEG. We begin with the optimal solution:

Lemma: $B_{OPT} \le \frac{x^2 n}{2} + o(x^2 n)$

Proof: The only back edges with respect to σ are the Type I and Type II edges. Thus, the number of Type I and Type II edges in σ are an upper bound for the number of back edges in the optimal solution. There are $(n-1)(x+(x-1)+\cdots+1) = \frac{x(x+1)(n-1)}{2}$ Type I edges, and $(n-1)\left(\sqrt{x}+\left(\sqrt{x}-1\right)+\cdots+1\right) = \frac{\sqrt{x}(\sqrt{x}+1)(n-1)}{2}$ Type II edges. The former is upper bounded by $\frac{x^2 n}{2}+\frac{xn}{2}=\frac{x^2 n}{2}+o(x^2 n)$. The latter is upper bounded by $\frac{xn}{2}+\frac{\sqrt{x}n}{2}=o(x^2 n)$. Thus, the total is $\frac{x^2 n}{2}+o(x^2 n)$ as desired.

Lemma: $B_{INCR} \geq \frac{5x^2n}{2} - o(x^2n)$

Proof: First, we show that all the vertices in b^i are ranked below all the vertices in $b^{i'}$ if $i < i'$. This is because from our earlier analysis of indegrees, we can find the maximum and minimum indegree for each block—

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

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

It follows that for all i, $max_i < min_{i+1}$ as desired. Thus all Type I and Type II edges are back edges; in the previous lemma, we found the number of such back edges to be at least $\frac{x^2n}{2} - o(x^2n)$.

Next, we count back edges within each block. We examine b^i for which $2 < i < n-1$. We can divide b^i into its left minigroups, $l_0, l_1, \cdots, l_{\sqrt{x}-1}$, and right minigroups, $r_0, r_1, \cdots, r_{\sqrt{x}-1}$ (where they are listed in order, so that l_0 is the leftmost minigroup and $r_{\sqrt{x}-1}$ is the rightmost minigroup). From our earlier analysis of indegrees, we have that the degree of a vertex in l_k is $(i-1)(2x+1) + x + \sqrt{x} - k$ and the degree of a vertex in r_k is $(i-1)(2x+1) + x - k - 1$. It follows that ordering by indegree gives $r_{\sqrt{x}-1}, \cdots, r_0, b_x^i, l_{\sqrt{x}-1}, \cdots, l_0$. Thus, aside from the edges within each minigroup, all the edges within b^i are back edges. This gives a lower bound of $\binom{2x+1}{2} = x(2x+1)$ edges in the block minus $2\sqrt{x}\binom{\sqrt{x}}{2} = x(\sqrt{x}-1)$ edges within minigroups. Since we were considering b^i for $2 < i < n-1$, this gives us an overall lower bound of $(n-4)\left(x(2x+1) - x(\sqrt{x}-1)\right) \geq 2x^2n - o(x^2n)$.

Thus if we combine these two parts, we have at least $\frac{5x^2n}{2} - o(x^2n)$ back edges, as desired.

From these two lemmas, it follows directly that the ratio between B_{INCR} and B_{OPT} has a limit of 5.

In summary, the primary contribution of this paper is an analysis of the INCR-INDEG method. The authors prove that it gives a 5-approximation for both weighted and unweighted FAS-tournament, and additionally prove that 5 is a tight bound. While this performance is not as good as other algorithms that have been proposed for FAS-tournament, INCR-INDEG is markedly simpler to implement. Furthermore, it is possible that this algorithm has a better performance guarantee for RANK-AGGREGATION, since the construction used for the tight bound does not translate to RANK-AGGREGATION. This holds potential for further study.