Linear Programming Based Approximation Algorithms for Feedback Set Problems in Bipartite Tournaments

Anke van Zuylen

Institute for Theoretical Computer Science, Tsinghua University, Beijing, China

Abstract. We consider the feedback vertex set and feedback arc set problems in bipartite tournaments. We improve on recent results by giving a 2-approximation algorithm for the feedback vertex set problem. We show that this result is the best we can attain when using a certain linear program as the lower bound on the optimal value. For the feedback arc set problem in bipartite tournaments, we show that a recent 4-approximation algorithm proposed by Gupta [5, 6] is incorrect. We give an alternative 4-approximation algorithm based on an algorithm for feedback arc set in (regular) tournaments in [10, 11].

1 Introduction

We consider the feedback vertex set problem and the feedback arc set problem on bipartite tournaments. The feedback vertex set problem on a directed graph $G = (V, A)$ asks for a set of vertices V' of minimum size such that the subgraph of G induced by $V \backslash V'$ is acyclic. The feedback arc set problem on G asks for a set of arcs A' of minimum size such that $(V, A \backslash A')$ is acyclic. A bipartite tournament is an orientation of a complete bipartite graph.

The feedback vertex set problem and the feedback arc set problem are equivalent on general directed graphs: given a directed graph $G = (V, A)$ we can create a graph G' which has a vertex for every arc in A, and an arc from vertex (u, v) to vertex (v, w). A directed cycle in G corresponds to a directed cycle in G' and vice versa; hence a feedback vertex set in G' corresponds to a feedback arc set in G, and a feedback arc set in G' corresponds to a feedback vertex set in G. The feedback arc/vertex set problem in general graphs is APX-hard [7], and can be approximated to within $\log |V| \log \log |V|$ [3, 9]. On bipartite tournaments, the problems are no longer equivalent, since if G is a bipartite tournament, then the graph G', as defined above, is bipartite but not necessarily a bipartite tournament and vice versa. However, both problems were shown to be NP-hard on bipartite tournaments as well [4, 2].

Cai, Deng and Zang [2] study a certain linear programming relaxation of the feedback vertex set problem in bipartite tournaments. They characterize certain small "forbidden subgraphs", and show that for an instance which does not contain such a subgraph, the linear program is totally dual integral: both the linear program and its dual have integer optimal solutions. Their work also implies a 3.5-approximation algorithm for the feedback vertex set problem in bipartite tournaments. Prashant [8] recently improved this result by giving a 3-approximation algorithm for feedback vertex set in bipartite tournaments. He uses the optimal solution to a linear programming relaxation of the feedback vertex set problem, and shows that one can iteratively round variables that are $\geq \frac{1}{3}$, until one obtains a feasible integer solution of cost at most 3 times the cost of the linear program. Gupta [5, 6] claims a (randomized) 4-approximation algorithm for feedback arc set in bipartite tournaments, by adapting the approach of Ailon, Charikar and Newman [1]. She shows that one can obtain a deterministic algorithm with the same guarantee, by using the optimal solution to a linear programming relaxation and the ideas in [10].

In this paper, we start by giving an alternative method for rounding the linear program for feedback vertex set in bipartite tournaments used by Prashant [8] which also gives an integer solution that costs at most 3 times the optimal value of the respective linear program. Our algorithm simply rounds up the variables that are at least $\frac{1}{2}$ plus all variables that are strictly greater than 0 and that correspond to vertices "on the left" in the bipartite tournament. Our algorithm and its analysis immediately suggest two improvements. First, we could also round up the variables that are strictly greater than 0 that correspond to vertices "on the right". We show that taking the better of these two rounded solutions yields a $\frac{5}{2}$-approximation algorithm. Our second improvement uses iterated rounding, where we solve the linear program, round up the variables that are greater than $\frac{1}{2}$, formulate a new linear program, and repeat. At some point, all variables are either 1 or less than $\frac{1}{2}$. Once this condition is reached, we show how to round the remaining solution and bound the cost against the dual solution to get a 2-approximation algorithm. We show that this result is tight for the linear program under consideration: we demonstrate an example with integrality gap 2, hence one cannot obtain a better approximation algorithm by using the lower bound given by the linear program.

Next, we consider the feedback arc set problem in bipartite tournaments. We point out a problem in the analysis of the algorithm used by Gupta [5, 6] and show that it does not give a constant factor approximation algorithm. However, we give another algorithm that does indeed obtain the result claimed by Gupta.

2 Feedback Vertex Set in Bipartite Tournaments

We are given a bipartite tournament $G = (V, A)$, and want to find a set of vertices $V' \subseteq V$ such that the subgraph of G induced by $V \backslash V'$ is acyclic, and $|V'|$ is minimal. We consider here a more general problem, in which for each $i \in V$, we are given a weight $w_i \geq 0$, and the goal is to find a feedback vertex set V' of minimum weight $\sum_{i \in V'} w_i$.

We use the following well known lemma [2, 8].

Lemma 1. *A bipartite tournament is acyclic if and only if it contains no cycle of length 4.*

Given a bipartite tournament $G = (V, A)$, let $\mathcal{C}$ be the set of cycles of length 4, i.e. $C \in \mathcal{C}$ is given by $\{i_1, (i_1, i_2), i_2, (i_2, i_3), i_3, (i_3, i_4), i_4, (i_4, i_1)\}$ with $i_1, \ldots, i_4 \in V$ and $(i_1, i_2), (i_2, i_3), (i_3, i_4), (i_4, i_1) \in A$. By Lemma 1, we have the following integer program for the feedback vertex set problem in a bipartite tournament:

$$\begin{aligned} & \min \sum_{i \in V} w_i x_i \\ \text{(FVS} - \text{BT)} \quad & \text{s.t.} \sum_{i \in C \cap V} x_i \geq 1, \forall C \in \mathcal{C} \\ & x_i \in \{0, 1\}, \forall i \in V. \end{aligned}$$

By solving the linear programming (LP) relaxation of this integer program, and rounding the values that are at least $\frac{1}{4}$, we can construct a solution with objective value of at most 4 times the optimal value. Prashant showed that in fact one can always find an optimal solution to the LP relaxation where some variable is at least $\frac{1}{3}$. Hence repeatedly rounding up these variables gives a 3-approximation algorithm.

We will begin by demonstrating another 3-approximation algorithm, where we bound the value of the solution against the dual of the LP relaxation, rather than the primal. Based on the ideas of this algorithm, we then show how to obtain an improved approximation algorithm.

The dual of the LP relaxation of (FVS-BT) is given by

$$\begin{aligned} \max \quad & \sum_{C \in \mathcal{C}} y_C \\ \text{s.t.} \quad & \sum_{C \in \mathcal{C}: i \in C} y_C \leq w_i, \forall i \in V \\ & y_C \geq 0, \forall C \in \mathcal{C}. \end{aligned}$$

Let $\{x_i\}_{i \in V}$ be an optimal solution to the linear relaxation. Let L, R be the partition of the vertices, so that all arcs in the bipartite tournament have one endpoint in L and one endpoint in R.

Lemma 2. *There exists a 3-approximation algorithm for feedback vertex set in bipartite tournaments.*

Proof. We create an integer solution $\hat{x}_i$ as follows: If $x_i \geq \frac{1}{2}$, or if $x_i > 0$ and $i \in L$ then $\hat{x}_i = 1$, else $\hat{x}_i = 0$. Note that $\{\hat{x}_i\}_{i \in V}$ is a feasible integer solution, since every cycle C has either some $i \in C$ such that $x_i \geq \frac{1}{2}$, or $|\{i \in C : x_i > 0\}| \geq 3$, in which case $\{i \in C : x_i > 0\} \cap L \neq \emptyset$.

Let $\{y_C\}_{C \in \mathcal{C}}$ be an optimal solution to the dual. We will need the following claim in our analyis:

Claim. Let $\{x_i\}_{i \in V}, \{y_C\}_{C \in \mathcal{C}}$ be optimal primal and dual solutions, and let $\hat{x}_i$ be given as above. Then for every $C \in \mathcal{C}$ either $|\{i \in C : \hat{x}_i = 1\}| \leq 3$ or $y_C = 0$.

Consider any $C \in \mathcal{C}$. If $|\{i \in C : \hat{x}_i = 1\}| > 3$, then every vertex in C has $\hat{x}_i = 1$. This means that $x_i > 0$ for $i \in C \cap L$, and $x_i \geq \frac{1}{2}$ for $i \in C \cap R$. But then $\sum_{i \in C} x_i > 1$ and by complementary slackness we know that $y_C = 0$. ◇

Note that if $\hat{x}_i = 1$, then $x_i > 0$, and by complementary slackness, we know that $\sum_{C \in \mathcal{C}: i \in C} y_C = w_i$. Therefore we get that

$$\begin{aligned}
\sum_{i \in V: \hat{x}_i = 1} w_i &= \sum_{i \in V: \hat{x}_i = 1} \sum_{C: i \in C} y_C \\
&= \sum_{C \in \mathcal{C}} y_C |\{i \in C : \hat{x}_i = 1\}| \\
&\leq 3 \sum_{C \in \mathcal{C}} y_C \\
&= 3 \sum_{i \in V} w_i x_i.
\end{aligned}$$

where the inequality follows from the claim. □

The algorithm and analysis in the proof of Lemma 2 suggest two ways of getting improved approximation guarantees. First of all, note that for i such that $0 < x_i < \frac{1}{2}$, the integer solution we created arbitrarily chose to set $\hat{x}_i = 1$, if $i \in L$; we could also have chosen to set $\hat{x}_i = 1$, if $i \in R$. Indeed, taking the better of these two solutions gives an improved approximation factor of 2.5, as we prove in Lemma 3. Secondly, instead of rounding up all variables on one side of the partition, we could only round up the variables that are at least $\frac{1}{2}$, and then resolve the linear program. In Lemma 4 we show that this gives a 2-approximation algorithm. Although we thus immediately improve the result from Lemma 3, we include Lemma 3 because it does not require us to solve linear programs repeatedly.

Lemma 3. *There exists a* 2.5*-approximation algorithm for feedback vertex set in bipartite tournaments.*

Proof. We define two solutions $\hat{x}_i^{(L)}$ and $\hat{x}_i^{(R)}$, where for $Z \in \{L, R\}$, we define $\hat{x}_i^Z$ to be 1 if $x_i \geq \frac{1}{2}$, or if $x_i > 0$ and $i \in Z$. By the arguments in the proof of Lemma 2, both $\{\hat{x}_i^{(L)}\}_{i \in V}$ and $\{\hat{x}_i^{(R)}\}_{i \in V}$ are feasible integer solutions.

Claim. Let $\{x_i\}_{i \in V}, \{y_C\}_{C \in \mathcal{C}}$ be optimal primal and dual solutions, and let $\hat{x}_i^{(Z)}$ for $Z = L, R$ be defined as above. Then for every $C \in \mathcal{C}$

$$|\{i \in C : \hat{x}_i^{(L)} = 1\}| + |\{i \in C : \hat{x}_i^{(R)} = 1\}| \leq 5 \text{ or } y_C = 0.$$

Consider any $C \in \mathcal{C}$. If $y_C > 0$, then $|\{i \in C : x_i \geq \frac{1}{2}\}| \leq 2$. We consider three cases:

(i) If $|\{i \in C : x_i \geq \frac{1}{2}\}| = 0$, then $|\{i \in C : \hat{x}_i^{(L)} = 1\}| \leq 2$ and $|\{i \in C : \hat{x}_i^{(R)} = 1\}| \leq 2$.

(ii) If $|\{i \in C : x_i \geq \frac{1}{2}\}| = 1$, suppose without loss of generality that there exists $i \in C \cap L$ such that $x_i \geq \frac{1}{2}$. Then $|\{i \in C : \hat{x}_i^{(L)} = 1\}| \leq 2$ and $|\{i \in C : \hat{x}_i^{(R)} = 1\}| \leq 3$.

(iii) If $|\{i \in C : x_i \geq \frac{1}{2}\}| = 2$, then by the fact that $y_C > 0$ and complementary slackness, we know that $\sum_{i \in C} x_i = 1$ and hence $|\{i \in C : x_i > 0\}| = 2$, so $|\{i \in C : x_i^{(L)} = 1\}| = 2$ and $|\{i \in C : \hat{x}_i^{(R)} = 1\}| = 2$.

$\diamond$

As before, if $\hat{x}_i^{(Z)} = 1$, then $x_i > 0$, and by complementary slackness, we know that $\sum_{C \in \mathcal{C}: i \in C} y_C = w_i$. So now we get that

$$\begin{aligned}
\sum_{i \in V: \hat{x}_i^{(L)}=1} w_i + \sum_{i \in V: \hat{x}_i^{(R)}=1} w_i &= \sum_{i \in V: \hat{x}_i^{(L)}=1} \sum_{C: i \in C} y_C + \sum_{i \in V: \hat{x}_i^{(R)}=1} \sum_{C: i \in C} y_C \\
&= \sum_{C \in \mathcal{C}} y_C \big(|\{i \in C : \hat{x}_i^{(L)} = 1\}| + |\{i \in C : \hat{x}_i^{(R)} = 1\}|\big) \\
&\leq 5 \sum_{C \in \mathcal{C}} y_C \\
&= 5 \sum_{i \in V} w_i x_i,
\end{aligned}$$

where the inequality follows from the claim. $\square$

Lemma 4. *There exists a 2-approximation algorithm for feedback vertex set in bipartite tournaments.*

Proof. Our algorithm solves the linear program $(\mathrm{FVS} - \mathrm{BT})$, rounds up the variables that are $\geq \frac{1}{2}$, remove the corresponding vertices from the graph and then resolves the linear program. If no variables $\geq \frac{1}{2}$ exist, we use the algorithm in the proof of Lemma 2 to complete the solution to a feasible integer solution.

Let V_k be the vertex set at the beginning of the k-th iteration of the algorithm, i.e. $V_1 = V$, and $V_k \subset V_{k-1}$ for $k \geq 2$. Let $G(V_k)$ be the induced bipartite tournament on V_k, let $OPT(V_k)$ be the optimal value of the LP on $G(V_k)$, and let $ALG(V_k)$ be the weight of the algorithm's solution restricted to V_k. Let ℓ be the total number of iterations of the algorithm.

We prove by backward induction on the algorithm that the variables rounded to 1 in iterations $k, \ldots, \ell$ give a feasible feedback vertex set on $G(V_k)$ of weight at most $2OPT(V_k)$. Since $OPT(V_1)$ is a lower bound on the value of the optimal feedback vertex set, this implies the lemma.

At the start of the last iteration, let $\mathcal{C}(V_\ell)$ be the 4-cycles in $G(V_\ell)$. Let $\{x_i^{(\ell)}\}_{i \in V_\ell}$ be an optimal primal, and $\{y_C^{(\ell)}\}_{C \in \mathcal{C}(\mathcal{V}_\ell)}$ be an optimal dual for the LP on this instance. Since $x_i^{(\ell)} < \frac{1}{2}$ for each $i \in V_\ell$, every $C \in \mathcal{C}(V_\ell)$ contains at least 3 vertices with strictly positive value $x_i^{(\ell)}$. Hence if we round up the variables for $i \in L$ with $x_i^{(\ell)} > 0$, we hit every cycle in $\mathcal{C}(V_\ell)$ at least once, and at most twice.

It follows that the solution we create is feasible on $G(V_\ell)$, and following the proof of Lemma 2, its weight is at most $2\sum_{C\in\mathcal{C}(V_\ell)} y_C = 2\sum_{i\in V_\ell} w_i x_i^{(\ell)} = 2OPT(V_\ell)$.

Now consider the beginning of iteration $k < \ell$. We solve the LP on $G(V_k)$, and let $\{x_i^{(k)}\}_{i\in V_k}$ be the optimal primal solution. The algorithm returns a feasible solution on $G(V_k)$: every 4-cycle either has a vertex i such that $x_i^{(k)} \geq \frac{1}{2}$, or it is a 4-cycle also in $G(V_{k+1})$, and by induction we know that our solution hits every 4-cycle in $G(V_{k+1})$.

By induction, $ALG(V_{k+1}) \leq 2OPT(V_{k+1})$. Note that

$$\begin{aligned} ALG(V_k) &= \sum_{i\in V_k : x_i^{(k)} \geq \frac{1}{2}} w_i + ALG(V_{k+1}) \\ &\leq 2 \sum_{i\in V_i : x_i^{(k)} \geq \frac{1}{2}} w_i x_i^{(k)} + 2OPT(V_{k+1}) \\ &= 2 \sum_{i\in V_i : x_i^{(k)} \geq \frac{1}{2}} w_i x_i^{(k)} + 2 \sum_{i\in V_{k+1}} w_i x_i^{(k+1)}. \end{aligned}$$

We note that $\{x_i^{(k)}\}_{i\in V_{k+1}}$ (the optimal LP solution on $G(V_k)$ restricted to V_{k+1}) is a feasible solution to the LP on $G(V_{k+1})$. Therefore $2\sum_{i\in V_{k+1}} w_i x_i^{(k+1)} \leq 2\sum_{i\in V_{k+1}} w_i x_i^{(k)}$, and since every vertex is either in V_{k+1}, or has $x_i^{(k)} \geq \frac{1}{2}$, we get that $ALG(V_k) \leq 2\sum_{i\in V_k} w_i x_i^{(k)} = 2OPT(V_k)$. $\square$

We conclude this section by showing that the result in Lemma 4 is the best one can hope for if using the optimal value of the LP relaxation of $(\mathrm{FVS}-\mathrm{BT})$ as a lower bound. The *integrality gap* of an integer linear program is the worst case ratio between the optimal value of the integer program and the optimal value of its LP relaxation, and hence a lower bound on the integrality gap implies a lower bound on the approximation ratio of an algorithm that bounds the cost of the algorithm's solution against the optimal value of the LP relaxation.

Lemma 5. *The integrality gap of* $(\mathrm{FVS}-\mathrm{BT})$ *is 2.*

Proof. By Lemma 4 the integrality gap is at most 2. We construct an example in which the integrality gap approaches 2. In particular, we will show that there exists an instance on $2n$ vertices for which the minimum feedback vertex set has size at least $n-1$. It remains to note that setting $x_i = \frac{1}{4}$ for all $i \in V$ always gives a feasible solution to $(\mathrm{FVS}-\mathrm{BT})$.

Let G_{2n} be a bipartite tournament with the following properties. We have vertices $L = \{\ell_1, \ldots, \ell_n\}$ and $R = \{r_1, \ldots, r_n\}$ and all arcs have one endpoint in L and one endpoint in R. In addition, we require that the arc between ℓ_i and r_i is directed from ℓ_i to r_i. We show by induction that there exists such a graph G_{2n} with minimum feedback vertex set of size at least $n-1$ for all $n \geq 2$.

For $n = 2$, G_4 is just a cycle of length 4. Given a graph G_{2n}, we construct $G_{2(n+1)}$ by adding a vertex ℓ_{n+1} to L and a vertex r_{n+1} to R, and we add the arc (ℓ_{n+1}, r_{n+1}), plus arcs (r_i, ℓ_{n+1}) and (r_{n+1}, ℓ_i) for every $i \leq n$. Note that the new arc (ℓ_{n+1}, r_{n+1}) is in a directed 4-cycle with every pair (ℓ_i, r_i). Hence a feedback vertex set in $G_{2(n+1)}$ either removes one of the new vertices ℓ_{n+1}, r_{n+1}, plus a minimum feedback vertex set in G_{2n} (thus removing at least n vertices) or it must remove one of ℓ_i, r_i for every $i \leq n$. Hence the size of the feedback vertex set on $G_{2(n+1)}$ is at least n. $\square$

3 Feedback Arc Set in Bipartite Tournaments

We now consider the feedback arc set problem in bipartite tournaments. Gupta [5, 6] recently gave an algorithm for this problem, and claimed it was a 4-approximation algorithm. We believe however that there is an error in the analysis. The algorithm is similar to that proposed by Ailon et al. [1] for the feedback arc set problem in tournaments and it recursively constructs an ordering of the vertices. The feedback arc set then consists of the *backward arcs*, i.e. the arcs going from right to left in the final ordering. To order the vertices, the proposed algorithm chooses an arc (i, j) as "pivot", and orders a vertex u to the left of the arc if either $(u, i) \in A$ or $(u, j) \in A$, and to the right of (i, j) otherwise. It then recurses on the two instances induced by the vertices on the left and on the right respectively. The key to their analysis, which we believe to be incorrect, is the claim that "an arc $(u, v) \in A$ becomes a backward arc if and only if $\exists (i, j) \in A$ such that (i, j, u, v) forms a directed 4-cycle in G and (i, j) was chosen as the pivot when all 4 were part of the same recursive call". Note, however, that an arc (u, v) may also become backward if $(i, u) \in A$ and $(v, j) \in A$, and (i, j) is chosen as the pivot when i, j, u, v were in the same recursive call. In that case, i, j, u, v are not in a directed 4-cycle, since we have $(i, u), (u, v), (v, j), (i, j) \in A$.

As an example, in the following instance, the optimal feedback arc set has size 1, and the expected number of backward arcs created by Gupta's algorithm is $O(n^2)$. We have vertices $L = \{\ell_1, \ldots, \ell_n\}$ and $R = \{r_1, \ldots, r_n\}$. We think of the vertices as being ordered as $\ell_1, r_1, \ell_2, r_2, \ldots, \ell_n, r_n$. All arcs have one endpoint in R and one endpoint in L and go from left to right except for the arc (r_n, ℓ_1). The number of backward arcs created in the first iteration of Guptas algorithm is $O(n^2)$ with constant probability: choosing an arc at random is the same as choosing a left vertex and a right vertex independently at random. Hence the probability that we choose (ℓ_i, r_j) with $1 < i \leq \frac{1}{4}n$ and $\frac{3}{4}n \leq j < n$ is $\frac{n/4-1}{n}\frac{n/4-1}{n} \approx \frac{1}{16}$. By pivoting on this arc, the arcs $(r_k, \ell_{k'})$ for $i \leq k \leq k' \leq j$ become backward, and since $i \leq \frac{1}{4}n, j \geq \frac{3}{4}n$, there are $O(n^2)$ such arcs.

We propose a more direct extension of the algorithm of Ailon et al. [1], or more precisely, we directly apply its derandomization by Van Zuylen et al. [11] (see also [10]) to the case of the feedback arc set problem in bipartite tournament. This allows us to obtain a 4-approximation algorithm.

We will use the following linear program. Let $x_{(i,j)} = 1$ denote that i is ordered before j. For an arc $(i, j) \in A$, let $w_{(i,j)} = 1$. If $(i, j) \notin A$, we let $w_{(i,j)} = 0$. Note that if i and j are not both in L or R, then $w_{(i,j)} + w_{(j,i)} = 1$, otherwise $w_{(i,j)} = w_{(j,i)} = 0$.

$$
\begin{aligned}
&\min \sum_{i<j} (w_{(j,i)} x_{(i,j)} + w_{(i,j)} x_{(j,i)}) \\
\text{(FAS)} \quad &\text{s.t. } x_{(i,j)} + x_{(j,k)} + x_{(k,i)} \geq 1, \forall \text{ distinct } i,j,k \\
&\qquad x_{(i,j)} + x_{(j,i)} = 1, \forall \text{ distinct } i,j \\
&\qquad x_{(i,j)} \geq 0, \forall \text{ distinct } i,j.
\end{aligned}
$$

The algorithm proposed in [11, 10] for the feedback arc set problem in *tournaments* (rather than bipartite tournaments) starts by solving the linear program (FAS). Based on the optimal solution, they form a tournament $T = (V, A_T)$, which has $(i,j) \in A_T$ if $x_{(i,j)} \geq \frac{1}{2}$ (where ties are broken arbitrarily if $x_{(i,j)} = x_{(j,i)} = \frac{1}{2}$).

The algorithm recursively constructs an ordering of the vertices, by choosing a pivot *vertex* k, ordering vertex i to the left of k if $(i,k) \in A_T$, and to the right of k if $(k,i) \in A_T$. It then recurses on the instances induced by the vertices on the left and right.

We can directly apply this algorithm to the feedback arc set problem in bipartite tournaments. Note that there are two types of backward arcs (from the original bipartite tournament) in the ordering constructed: arcs $(i,j) \in A$ for which $(j,i) \in A_T$ (i.e. $x_{(j,i)} \geq \frac{1}{2}$) and one of i,j is chosen as pivot when i,j are in the same recursive call, and arcs $(i,j) \in A$ for which there exists k such that $(j,k) \in A_T, (k,i) \in A_T$ and k is chosen as pivot when i,j,k are in the same recursive call.

Clearly, we can bound the cost of the first type of backward arcs against twice the contribution of (i,j) to the linear program's objective value. In order to bound the cost of the second type of backward arcs, [11, 10] chooses a pivot carefully. Let $T_k(V)$ denote the pairs (i,j) such that $(j,k) \in A_T$ and $(k,i) \in A_T$. In a recursive call with vertex set V, the pivot k is the vertex that minimizes

$$
\frac{\sum_{(i,j) \in T_k(V)} w_{(i,j)}}{\sum_{(i,j) \in T_k(V)} (w_{(j,i)} x_{(i,j)} + w_{(i,j)} x_{(j,i)})}.
$$

It follows from the Theorem 2.1 in Van Zuylen and Williamson [11] that if the following condition holds for every $(i,j), (j,k), (k,i) \in A_T$, then it is always possible to choose a pivot k such that the above ratio is at most 4.

$$
\begin{aligned}
w_{(i,j)} + w_{(j,k)} + w_{(k,i)} \leq 4 \Big(&w_{(j,i)} x_{(i,j)} + w_{(i,j)} x_{(j,i)} + \\
&+ w_{(k,j)} x_{(j,k)} + w_{(j,k)} x_{(k,j)} + w_{(i,k)} x_{(k,i)} + w_{(k,i)} x_{(i,k)} \Big). \qquad (1)
\end{aligned}
$$

Hence we can bound the cost of the second type of backward arcs against 4 times their contribution to the linear program's objective value. It thus follows the algorithm is a 4-approximation algorithm.

Lemma 6. *There exists a 4-approximation algorithm for feedback arc set in bipartite tournaments.*

Proof. We need to show that (1) holds. Note that for any triple such that $(i,j),(j,k),(k,i) \in A_T$, it must either be the case that all three vertices were on the same side of the bipartite tournament G, or exactly two were on one side, and the other vertex was on the other side. In the first case, the left hand side of (1) is 0 and there is nothing to prove. In the second case, suppose without loss of generality that $w_{(i,j)} = w_{(j,i)} = 0$.

We need to show that

$$w_{(j,k)} + w_{(k,i)} \leq 4\Big(w_{(k,j)}x_{(j,k)} + w_{(j,k)}x_{(k,j)} + w_{(i,k)}x_{(k,i)} + w_{(k,i)}x_{(i,k)}\Big).$$

We rewrite the right hand side as $4\Big((1 - w_{(j,k)})x_{(j,k)} + w_{(j,k)}(1 - x_{(j,k)}) + (1 - w_{(k,i)})x_{(k,i)} + w_{(k,i)}(1 - x_{(k,i)})\Big) = 4\Big(w_{(j,k)} + w_{(k,i)} - x_{(j,k)}(2w_{(j,k)} - 1) - x_{(k,i)}(2w_{(k,i)} - 1)\Big)$.

Note that $x_{(j,k)} \geq \frac{1}{2}, x_{(k,i)} \geq \frac{1}{2}$ and $x_{(i,j)} \geq \frac{1}{2}$ by the fact that $(j,k),(k,i),(i,j) \in A_T$. Hence the right hand side is non-increasing in $w_{(k,i)}$ and $w_{(j,k)}$, and since the left hand side is increasing in $w_{(k,i)}$ and $w_{(j,k)}$, it is enough to consider the case when $w_{(k,i)} = w_{(j,k)} = 1$. It thus remains to show that $4(2 - x_{(j,k)} - x_{(k,i)}) \geq 2$.

By the second set of constraints of (FAS), $4(2 - x_{(j,k)} - x_{(k,i)}) = 4(x_{(k,j)} + x_{(i,k)}$, and by the first set of constraints, $x_{(i,k)} + x_{(k,j)} \geq 1 - x_{(j,i)} = x_{(i,j)} \geq \frac{1}{2}$, which directly gives the desired inequality. □

We leave open the question of whether there exists a combinatorial algorithm that achieves the same guarantee. The idea of Gupta's algorithm to pivot on an *arc* of the graph, rather than a vertex as in Ailon et al. [1] is interesting, and it may be possible to modify the algorithm so that it does achieve a constant approximation guarantee.

References

1. Nir Ailon, Moses Charikar, and Alantha Newman, *Aggregating inconsistent information: ranking and clustering*, STOC '05: Proceedings of the 37th Annual ACM Symposium on Theory of Computing, 2005, pp. 684–693.
2. Mao-Cheng Cai, Xiaotie Deng, and Wenan Zang, *A min-max theorem on feedback vertex sets*, Math. Oper. Res. **27** (2002), 361–371.
3. Guy Even, Joseph Naor, Baruch Schieber, and Madhu Sudan, *Approximating minimum feedback sets and multicuts in directed graphs*, Algorithmica **20** (1998), no. 2, 151–174.
4. Jiong Guo, Falk Hüffner, and Hannes Moser, *Feedback arc set in bipartite tournaments is NP-complete*, Inf. Process. Lett. **102** (2007), no. 2-3, 62–65.
5. Sushmita Gupta, *Feedback arc set problem in bipartite tournaments*, TAMC '07: 4th International Conference on Theory and Applications of Models of Computation, Lecture Notes in Computer Science, vol. 4484, Springer, 2007, pp. 354–361.
6. Sushmita Gupta, *Feedback arc set problem in bipartite tournaments*, Inf. Process. Lett. **105** (2008), no. 4, 150–154.

7. Viggo Kann, *On the approximability of NP-complete optimization problems*, Ph.D. thesis, Department of Numerical Analysis and Computing Science, Royal Institute of Technology, Stockholm, 1992.
8. Prashant Sasatte, *Improved approximation algorithm for the feedback set problem in a bipartite tournament*, Operations Research Letters **36** (2008), no. 5, 602 – 604.
9. Paul D. Seymour, *Packing directed circuits fractionally*, Combinatorica **15** (1995), no. 2, 281–288.
10. Anke van Zuylen, Rajneesh Hegde, Kamal Jain, and David P. Williamson, *Deterministic pivoting algorithms for constrained ranking and clustering problems*, SODA '07: Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms, 2007, pp. 405–414.
11. Anke van Zuylen and David P. Williamson, *Deterministic pivoting algorithms for constrained ranking and clustering problems*, Math. Oper. Res. (to appear) `http://www.itcs.tsinghua.edu.cn/~anke/MOR.pdf`.