Feedback Arc Set in Bipartite Tournaments is NP-Complete

Jiong Guo [1] Falk Hüffner [1] Hannes Moser [2]

Institut für Informatik, Friedrich-Schiller-Universität Jena, Ernst-Abbe-Platz 2, D-07743 Jena, Germany

Abstract

The Feedback Arc Set problem asks whether it is possible to delete at most k arcs to make a directed graph acyclic. We show that Feedback Arc Set is NP-complete for bipartite tournaments, that is, directed graphs that are orientations of complete bipartite graphs.

Key words: combinatorial problems, computational complexity, feedback set problems, bipartite tournaments

Given a directed graph $G = (V, A)$ with vertex set V and arc set A, a feedback vertex (or arc) set is a subset of vertices (or arcs) that meets all cycles in G. The Feedback Vertex/Arc Set (FVS/FAS) problems ask to decide, for a given graph G and a nonnegative integer k, whether there is a feedback vertex/arc set of size at most k. Both problems are known to be NP-complete if we put no restriction on the input directed graphs [3].

Motivated by the general hardness results, many subclasses of directed graphs have been considered. It turns out that both FVS and FAS can be solved in polynomial time in reducible flow graphs [10] and in cyclically reducible flow graphs [12]. Due to important applications, e.g. in voting systems [7], feedback set problems restricted to tournaments received considerable attention. A tournament is a directed graph where there is exactly one arc between each

Email addresses: guo@minet.uni-jena.de (Jiong Guo), hueffner@minet.uni-jena.de (Falk Hüffner), moser@minet.uni-jena.de (Hannes Moser).

[1] Supported by the Deutsche Forschungsgemeinschaft, Emmy Noether research group PIAF (fixed-parameter algorithms), NI 369/4.

[2] Supported by the Deutsche Forschungsgemeinschaft, project ITKO (iterative compression for solving hard network problems), NI 369/5.

pair of vertices. Speckenmeyer [11] showed the NP-completeness of FVS in tournaments. FAS in tournaments was conjectured to be NP-hard for a long time [4], and only recently was this proven. Ailon, Charikar, and Newman [1] gave a randomized reduction from FAS on general directed graphs, which was independently derandomized in two works [2,6]. Independently, Conitzer [7] gave a deterministic reduction from MaxSAT.

FVS and FAS have also been studied for bipartite tournaments. A bipartite tournament is an orientation of a complete bipartite graph. Cai, Deng, and Zang [5] showed that FVS in bipartite tournaments is NP-complete. They have also established a min-max theorem for FVS in bipartite tournaments. Concerning FAS in bipartite tournaments, Gutin and Yeo [9] note that it is fixed-parameter tractable, that is, the exponential part of the running time can be restricted to the parameter k, the size of the feedback arc set. Dom et al. [8] gave a concrete algorithm solving FAS in bipartite tournaments in $O(3.38^k \cdot |V|^{O(1)})$ time. However, the complexity of FAS in bipartite tournaments was left open.

We close this gap here by showing that FAS is indeed NP-complete in bipartite tournaments. To this end, we give a polynomial-time many-one reduction from the CNF Satisfiability (CNF-SAT) problem. Our reduction is inspired by Conitzer's proof of NP-completeness of FAS in tournaments [7]. However, the observations used in his proof do not hold for bipartite tournaments, thus requiring a fairly different reduction. Interestingly, Cai, Deng, and Zang [5] also observe in their NP-completeness proof for FVS in bipartite tournaments that it seems to be a formidable (if not impossible) task to directly adapt a reduction for the non-bipartite case to the more complicated bipartite case. Note that our reduction also shows that FAS is NP-complete for c-partite tournaments for any fixed $c \geq 2$.

A *bipartite tournament* is an orientation of a complete bipartite graph, meaning its vertex set is the union of two disjoint sets V_1 and V_2 and there is exactly one arc between each pair of a V_1-vertex and a V_2-vertex. The Feedback Arc Set in bipartite tournaments (FASBT) problem is defined as follows:

> **Input**: A bipartite tournament G and a nonnegative integer k.
> **Task**: Find a *feedback arc set* S of at most k arcs whose removal from G results in an acyclic digraph.

In the following, we use the equivalent characterization of FASBT as the problem of finding a sort $\prec$ of the vertices of a bipartite tournament such that there are at most k *backward arcs*, that is, arcs $u \to v$ with $v \prec u$.[3]

[3] Note that a directed graph is acyclic iff it has a *topological sort*, that is, a sort without backward arcs.

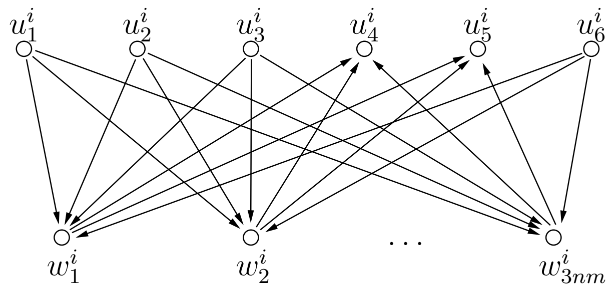

Fig. 1. The gadget X^i for variable x_i.

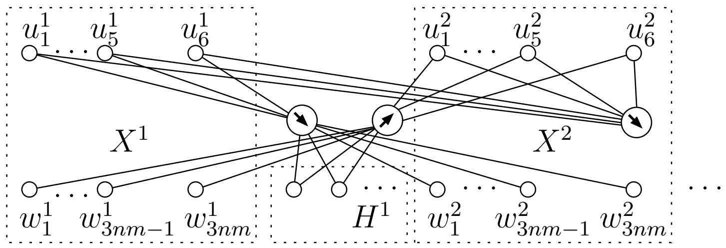

Fig. 2. Interconnection between two gadgets. For clarity, only a few vertices and edges are shown. The arrows in the circles indicate the direction of all-pairwise arcs.

Theorem 1 FEEDBACK ARC SET *in bipartite tournaments* (FASBT) *is* NP*-complete.*

PROOF. FASBT is clearly in NP. We prove the hardness by giving a reduction from CNF-SAT. Let F be a boolean formula in conjunctive normal form. Let $C := \{C_i : 1 \leq i \leq m\}$ be the set of clauses in F, and let $X := \{x_i : 1 \leq i \leq n\}$ be the set of variables in F.

We construct a bipartite tournament $G = (V, A)$, whose vertex set consists of four disjoint subsets, namely U-vertices, W-vertices, H-vertices, and clause vertices. For each variable x_i in F we use a variable gadget X^i consisting of six U-vertices $U^i := \{u_1^i, \ldots, u_6^i\}$ and $3nm$ W-vertices $W^i := \{w_j^i : 1 \leq j \leq 3nm\}$. From each vertex in $\{u_1^i, u_2^i, u_3^i, u_6^i\}$ we draw an arc to each vertex in W^i, and from each vertex in W^i we draw an arc to each of u_4^i and u_5^i (see Figure 1).

Then, we insert arcs between the variable gadgets. For every pair of variables x_i, x_j with $i < j$, we draw an arc from each vertex in U^i to each vertex in W^j and draw an arc from each vertex in W^i to each vertex in U^j.

Furthermore, we insert $n-1$ sets of H-vertices, $H^i := \{h_j^i : 1 \leq j \leq 3nm\}$ for each $1 \leq i \leq n-1$. These H-vertices are connected to the U-vertices of the variable gadgets: For every j with $1 \leq j \leq i$, we draw an arc from each vertex in U^j to each vertex in H^i. For every j with $i < j \leq n$, we draw an arc from each vertex in H^i to each vertex in U^j (see Figure 2).

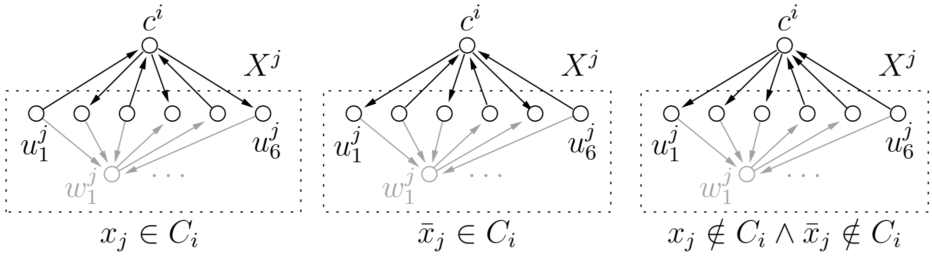

Fig. 3. The arcs between clause vertex c^i and variable gadget X^j.

Finally, for each clause C_i, we insert a clause vertex c^i and draw arcs between c^i and the U-vertices of each variable gadget as depicted in Figure 3.

The graph $G = (V, A)$ is clearly a bipartite tournament with partitions $V_1 = \bigcup_{1 \le i \le n} U^i$ and $V_2 = (\bigcup_{1 \le i \le n} W^i) \cup (\bigcup_{1 \le i \le n-1} H^i) \cup \{c^1, \dots, c^m\}$ and can be constructed in polynomial time.

Next, we show that F is satisfiable iff there is a feedback arc set of size at most $3nm - 2m$ for G, that is, we can find a sort of the vertices of G such that there are at most $3nm - 2m$ backward arcs.

"$\Rightarrow$": Given a subset $A \subseteq V$, we use $r(A)$ to denote an arbitrary sort of the vertices in A. First, we observe that G without any clause vertex is acyclic, that is, there is a topological sort $\prec$ of the vertices in $V \setminus \{c^i : 1 \le i \le m\}$:

$$s(X^1) \prec r(H^1) \prec s(X^2) \prec r(H^2) \prec \cdots \prec r(H^{n-1}) \prec s(X^n),$$

where, for $1 \le i \le n$,

$$s(X^i) := r(\{u_1^i, u_2^i, u_3^i, u_6^i\}) \prec r(W^i) \prec r(\{u_4^i, u_5^i\}).$$

Next, we show that if F is satisfiable, then we can extend $\prec$ to all vertices in G (thus, including the clause vertices c^i) such that we obtain a sort with at most $3nm - 2m$ backward arcs.

Suppose that we are given a truth assignment T satisfying F. The value assigned to variable x_i is denoted by $T(x_i)$. For each variable x_i, we create a set P_i to store the clauses that can be satisfied by $T(x_i)$. If a clause is satisfied by the assignments of at least two variables, then we add it only to one of the corresponding lists. We use Q_i to denote the set of the clause vertices whose corresponding clauses are in P_i. If $T(x_i)$ is true, then we modify $s(X^i)$ as follows:

$$s(X^i) := r(\{u_1^i, u_3^i\}) \prec r(Q_i) \prec r(\{u_2^i, u_6^i\}) \prec r(W^i) \prec r(\{u_4^i, u_5^i\});$$

otherwise, that is, $T(x_i)$ is false, we modify $s(X^i)$ as follows:

$$s(X^i) := r(\{u_2^i, u_6^i\}) \prec r(Q_i) \prec r(\{u_1^i, u_3^i\}) \prec r(W^i) \prec r(\{u_4^i, u_5^i\}).$$

We claim that the extended sort has $3nm - 2m$ backward arcs: Clearly, every backward arc has a clause vertex as one of its endpoints. Since T satisfies F, for each clause C_i, there exists a j, $1 \leq j \leq n$, with $C_i \in P_j$. Then, we can count one backward arc between the vertices in U^j and c^i, namely, $u_5^j \to c^i$ if $T(x_j)$ is true or $u_4^j \to c^i$ if $T(x_j)$ is false. For each variable gadget X^l with $l \neq j$, we count three backward arcs between the vertices in U^l and c^i. Thus, there are $3(n-1)+1$ backward arcs for each clause C_i, and thus there are $m \cdot (3(n-1)+1) = 3nm - 2m$ backward arcs.

"$\Leftarrow$": Let D be a feedback arc set for G of size at most $3nm - 2m$. Let G' be the graph that results from G if we remove the feedback arc set D.

First, we claim that in G', for any clause vertex c^i, there is at most one variable gadget that contains more than three neighbors[4] of c^i. To show this, we suppose for a contradiction that there are a clause vertex c^i and two variable gadgets X^j, X^k such that c^i has more than three neighbors in each of X^j and X^k. Assume that $j < k$. Note that in G, there are exactly three arcs from c^i to the U-vertices of a variable gadget and exactly three arcs from the U-vertices of a variable gadget to c^i. Since there are more than three neighbors of c^i in X^j, there exists in G' an arc from c^i to some U^j-vertex. For the same reason, there exists in G' an arc from some U^k-vertex to c^i. Since there are at least $3nm$ directed paths from every U^j-vertex to every U^k-vertex via vertices in H^j and $|D| \leq 3nm - 2m$, there is then a cycle in G' (via a vertex in H^j), which contradicts the acyclicity of G'.

With this claim, in G', for each vertex c^i there is at most one variable gadget in which c^i has more than three neighbors. Also, five neighbors are the maximum, since six neighbors would imply a cycle (in Figure 3 we can observe cycles through c^i via the black and gray arcs). Therefore, there are at most $3(n-1)+5 = 3n+2$ arcs in G' between a clause vertex c^i and all variable gadgets, thus in total G' has at most $3nm+2m$ arcs between all clause vertices and all variable gadgets. However, in G there are $6nm$ arcs between the clause vertices and the variable gadgets. Thus, we have to remove at least $6nm-(3nm+2m) = 3nm - 2m$ arcs from G to destroy all cycles. Since $|D| \leq 3nm - 2m$, we know that $|D| = 3nm - 2m$. This means that D contains only arcs incident to clause vertices, and, for each clause vertex c^i, there must be a variable gadget X^j in which c^i has exactly five neighbors in G'; we say that c^i is *linked* to X^j.

Furthermore, let C_j and C_k denote two clauses in F such that variable x_i occurs positively in C_j and negatively in C_k. Observe that, in G', at most one of the clause vertices c^j and c^k can be linked to the variable gadget X^i; otherwise, G' would contain a cycle (see Figure 4 for an illustration). Moreover, a clause vertex c^i cannot be linked to a variable gadget X^j whose corresponding

[4] Two vertices are neighbors if they are adjacent in the underlying undirected graph.

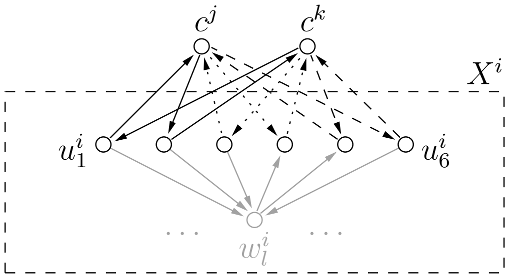

Fig. 4. To link both c^j and c^k to X^i would mean to delete two arcs (there are 12 arcs in total and "to link" means to preserve exactly five arcs per clause). However, there are three arc-disjoint cycles (dashed, dotted, and solid black arcs), thus we would have to delete at least three arcs.

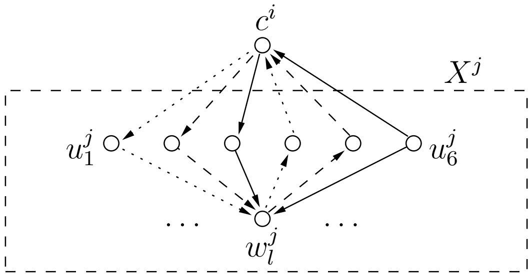

Fig. 5. There are two arc-disjoint cycles (dashed and dotted arcs), thus we could only preserve four arcs, but not five. Note that the $3nm$ W-vertices in each variable gadget are needed to assure that all cycles of this kind (from all clause vertices) are arc-disjoint, that is, that we cannot destroy many such cycles by just deleting a few arcs between the U-vertices and the W-vertices.

variable x_j does not occur in clause C_i, as this would also mean that G' contains a cycle (see Figure 5).

Altogether, in G' each clause vertex c^i is linked to exactly one variable gadget X^j and the corresponding variable x_j occurs in the corresponding clause C_i. If more than one clause vertex is linked to a variable gadget X^i, then variable x_i occurs in all of the corresponding clauses either positively or negatively, but not both. Thus, we can compute the following truth assignment: If there is a clause vertex c^i linked to a variable gadget X^j, then we set the corresponding variable x_j to true, if x_j occurs in C_i positively, and to false if x_j occurs in C_i negatively. Clearly, since all clause vertices c^i are linked to vertex gadgets, the computed truth assignment satisfies the formula F. □

Acknowledgements We thank Rolf Niedermeier for giving valuable comments and suggestions to improve the presentation of this paper.

References

[1] N. Ailon, M. Charikar, and A. Newman. Aggregating inconsistent information: ranking and clustering. In *Proc. 37th STOC*, pages 684–693. ACM Press, 2005.

[2] N. Alon. Ranking tournaments. *SIAM Journal on Discrete Mathematics*, 20(1):137–142, 2006.

[3] G. Ausiello, A. D'Atri, and M. Protasi. Structure preserving reductions among convex optimization problems. *Journal of Computer and System Sciences*, 21:136–153, 1980.

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

[5] M.-C. Cai, X. Deng, and W. Zang. A min-max theorem on feedback vertex sets. *Mathematics of Operations Research*, 27(2):361–371, 2002.

[6] P. Charbit, S. Thomassé, and A. Yeo. The minimum feedback arc set problem is NP-hard for tournaments. *Combinatorics, Probability and Computing*, 2006. To appear.

[7] V. Conitzer. Computing Slater rankings using similarities among candidates. In *Proc. 21st AAAI*. AAAI Press, 2006.

[8] M. Dom, J. Guo, F. Hüffner, R. Niedermeier, and A. Truß. Fixed-parameter tractability results for feedback set problems in tournaments. In *Proc. 6th CIAC*, volume 3998 of *LNCS*, pages 320–331. Springer, 2006.

[9] G. Gutin and A. Yeo. Some parameterized problems on digraphs. Submitted to a special issue of *The Computer Journal* on parameterized complexity, 2006.

[10] A. Shamir. A linear time algorithm for finding cutsets in reduced graphs. *SIAM Journal on Computing*, 8(4):645–655, 1979.

[11] E. Speckenmeyer. On feedback problems in digraphs. In *Proc. 15th WG*, volume 411 of *LNCS*, pages 218–231. Springer, 1989.

[12] C.-C. Wang, E. L. Lloyd, and M. L. Soffa. Feedback vertex sets and cyclically reducible graphs. *Journal of the ACM*, 32(2):296–313, 1985.