Parameterized Algorithms for Feedback Set Problems and Their Duals in Tournaments ⋆

Venkatesh Raman *, Saket Saurabh

The Institute of Mathematical Sciences, Chennai-600 113, India

Abstract

The parameterized feedback vertex (arc) set problem is to find whether there are k vertices (arcs) in a given graph whose removal makes the graph acyclic. The parameterized complexity of this problem in general directed graphs is a long standing open problem. We investigate the problems on tournaments, a well studied class of directed graphs. We consider both weighted and unweighted versions.

We also address the parametric dual problems which are also natural optimization problems. We show that they are fixed parameter tractable not just in tournaments but in oriented directed graphs (where there is at most one directed arc between a pair of vertices). More specifically, the dual problem we show fixed parameter tractable are: Given an oriented directed graph, is there a subset of k vertices (arcs) that forms an acyclic directed subgraph of the graph?

Our main results include:

- an $O((2.4143)^k n^\omega)$ [1] algorithm for weighted feedback vertex set problem, and an $O((2.415)^k n^\omega)$ algorithm for weighted feedback arc set problem in tournaments;
- an $O((e2^k/k)^k k^2 + \ min\{m \lg n, n^2\})$ algorithm for the dual of feedback vertex set problem (maximum vertex induced acyclic graph) in oriented directed graphs, and an $O(4^k k + m)$ algorithm for the dual of feedback arc set problem (maximum arc induced acyclic graph) in general directed graphs.

We also show that the dual of feedback vertex set is $W[1] - hard$ in general directed graphs and the feedback arc set problem is fixed parameter tractable in dense directed graphs. Our results are the first non trivial results for these problems.

Key words: tournaments, feedback vertex set, feedback arc set, parameterized complexity

[1] ω is the exponent of the best matrix multiplication algorithm.

1 Introduction and Motivation

Given a directed graph on n vertices and an integer parameter k, the feedback vertex (arc) set problem is to determine whether the given graph has a set of k vertices (arcs) whose removal results in an acyclic directed graph. In the weighted version of the problem we are given non negative weights on vertices (arcs) and the problem asks whether the graph has a set of vertices (arcs) of weight at most k, whose removal makes the graph acyclic. While these problems in undirected, unweighted graphs are known to be fixed parameter tractable [15,16] (the edge version in undirected graphs can be trivially solved), the parameterized complexity of these problems in directed graphs is a long standing open problem in the area. In fact, there are problems on sequences and trees in computational biology, that are related to the directed feedback vertex set problem [7].

In this paper, we consider these problems in the well studied special class of directed graphs, tournaments. A tournament $T = (V, E)$ is a directed graph in which there is exactly one directed arc between every pair of vertices. Feedback vertex set problem is NP-complete in tournaments [19]. Feedback arc set problem is not known to be (but conjectured to be) NP-complete in unweighted tournaments while it is NP-complete for weighted tournaments [5]. We give efficient fixed parameter tractable algorithms for the feedback vertex set and feedback arc set problem in weighted tournaments.

Weighted feedback arc set problem in tournaments finds application in rank aggregation methods. Dwork et.al [5] have shown that the problem of computing the so called Kemeny optimal permutation for k full lists, where k is an odd integer, is reducible to the problem of computing a minimum feedback arc set problem on a weighted tournament with weights between 1 and $k-2$.

Since a tournament has a directed cycle if and only if it has a directed triangle [1], the feedback vertex set problem in a tournament is a set of vertices that hits all the triangles in the tournament. Hence one can first find a directed triangle in the tournament, and then branch on each of its three vertices to get an easy recursive $O(3^k n^3)$ algorithm to find a feedback vertex set of size at most k (or determine its absence). This algorithm generalises for the weighted feedback vertex set problem with weights at least 1. We can also write the unweighted feedback vertex set problem in tournaments as a 3-hitting set problem (hitting set of all directed triangles) and can apply the algorithm of [13] to get an $O(2.27^k + n^3)$ algorithm. However, this algorithm uses some

⋆ Preliminary versions of this paper appeared in [17] and [18].

∗ Corresponding Author
Email addresses: vraman@imsc.res.in (Venkatesh Raman),
saket@imsc.res.in (Saket Saurabh).

preprocessing rules which don't naturally generalize for the weighted hitting set problem. In section 2, we give improved algorithms for the weighted version of the feedback vertex set problem in tournaments. We consider the following variants of the weighted feedback vertex set (WFVS) problem:

(1) Integer-WFVS, where the weights are arbitrary positive integers,
(2) Real-WFVS, where the weights are real numbers ≥ 1, and
(3) General-WFVS, where the weights are positive real numbers.

We show that the Integer-WFVS in a directed graph can be solved as fast as feedback vertex set problem in an unweighted directed graph. Since the reduction here preserves the tournament structure, Integer-WFVS can be solved as fast as the feedback vertex set problem in an unweighted tournament, which currently has running time of $O((2.27)^k + n^3)$ [13]. We show that Real-WFVS can be solved in $O((2.4143)^k n^\omega)$ time and that General-WFVS is not fixed parameter tractable (FPT) unless $P = NP$.

There are parameterized reductions between feedback vertex set problem (FVS) and feedback arc set problem (FAS) in weighted directed graphs (actually the reductions to show NP-complete for these problems are parameterized reductions [6]), but they don't preserve the tournament structure. Also, it is not sufficient to hit all triangles by arcs to get a feedback arc set in a tournament (for example, in the tournament in figure 2, $S = \{\{1,3\},\{4,2\}\}$ hits all the triangles, but S does not hit the cycle $\{1,2,3,4\}$). Furthermore, after we remove an arc from a tournament, we no longer have a tournament. Hence it is not straightforward to apply the ideas of the fixed parameter tractable algorithms for feedback vertex set to the arc set problem. In Section 3, we give three different algorithms for the FAS problem. We first develop an $O(\sqrt{k}^k n^\omega \log n)$ time algorithm for FAS using the fact that a directed graph G with at most k arcs away from a tournament T (i.e., T can be obtained from G by adding at most k arcs to it) has a cycle of length at most $O(\sqrt{k})$. In Section 3.2, we first show that if a subset F of arcs forms a minimal feedback arc set in a directed graph then the graph formed after reversing these arcs is acyclic. Such a characterization helps us to maintain the tournament structure (since in every recursive step we reverse but not delete arcs). We apply this characterization to develop an algorithm for FAS in tournaments taking $O(3^k n^\omega)$ time. We then improve this by using a branching technique to obtain an $O((2.415)^k n^\omega)$ time algorithm. We observe that the algorithm, and hence the bound, applies for the FAS problem in weighted tournaments as well, where weights on the arcs are at least 1. In Section 4, we show that FAS is fixed parameter tractable even for dense directed graphs (graphs having at least $\binom{n}{2} - n^{1+o(1)}$ arcs).

In Section 5, we consider the parametric duals of feedback set problems in directed graphs. More specifically the dual problems are : Given a directed graph G, (a) is there a set of at least k vertices of G that induces a directed acyclic graph, and (b) is there a directed acyclic subgraph of G with at least k arcs? In undirected graphs, the former ((a)) question is $W[1]$-complete [11] while the latter question is easily solvable in polynomial time (since in any connected graph on n vertices and m edges, it is necessary and sufficient to remove $m-(n+1)$ edges to make it acyclic).

In directed graphs where cycles of length 2 are allowed, we show that the parametric dual of the feedback vertex set problem is $W[1]$-hard, while it is fixed parameter tractable for oriented directed graphs (where cycles of length 2 are not allowed). We show that the dual of the feedback arc set problem is fixed parameter tractable in general directed graphs. We also consider variations of these problems where the parameter is above the default lower bound.

In section 6, we conclude with some remarks and open problems. Throughout this paper, by $\log n$ and ω, we mean, respectively, the logarithm to the base 2 of n and the exponent of the running time of the best matrix multiplication algorithm. By $rev(x)$, where $x = (u, v)$ is an arc of a directed graph, we mean the arc (v, u). By an oriented directed graph, we mean a directed graph where there is at most one directed arc between every pair of vertices. By an inneighbour of a vertex x in a directed graph G, we mean a vertex y such that there is a directed arc from y to x in G. An outneighbour of a vertex is similarly defined.

2 Feedback Vertex Set Problem in Tournaments

We recall that by writing feedback vertex set in tournaments as a hitting set of all triangles, we have the following using [13] :

Lemma 1 *[13] Given a tournament $T = (V, E)$, we can determine whether T has a feedback vertex set of size at most k in $O((2.27)^k + n^3)$ time.*

2.1 Integer-WFVS

Integer-WFVS is a variant of weighted feedback vertex set problem, where weights are arbitrary positive integers.

Theorem 1 *There exists a parameterized many-one reduction from* Integer-WFVS *to unweighted feedback vertex set in directed graphs.*

Proof: Let G be an integer weighted directed graph. Any vertex having weight strictly more than k can not be a part of any minimal feedback vertex set of weight at most k. So given the weight function π if some vertex v has $\pi(v) > k$

then we make $\pi(v) = k + 1$. It is easy to see that G has a feedback vertex set of weight at most k if and only if it has a feedback vertex set of weight at most k with the modified weight function.

We will construct a new directed graph G' from G as follows: replace each vertex v having weight $\pi(v) = w > 1$ with a cluster V' consisting of w vertices. If there is an arc (u, v) in the original graph G then we add an arc from every vertex of the cluster U' to every vertex in V'. For every cluster V', we add intra cluster arcs such that $G[V']$ is a transitive tournament. Here $G[V']$ represents the induced directed graph on V'.

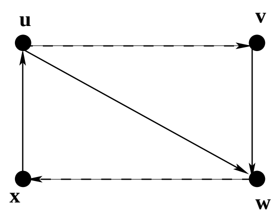

Fig. 1. A Witness Cycle

We claim that G has a feedback vertex set (FVS) of weight at most k if and only if G' has a FVS of size at most k. Let $\{v_1, v_2, \cdots, v_l\}$ be a FVS of weight at most k in G. Then the vertices of the corresponding clusters $\{V_1', V_2', \cdots, V_l'\}$ form a FVS of size at most k in G'. The other direction follows from the observation that every minimal feedback vertex set (F) of size at most k in G' has either all the vertices of any cluster or none of them. To see this, assume that there is a cluster V' such that there is a vertex $v \in V'$ in F and a $u \in V'$ not in F. Since $v \in F$, and F is minimal, there exists a witness cycle C such that $F \cap C = \{v\}$. Now if $u \notin C$ then we get a cycle C' in T' by replacing v with u in C such that $C' \cap F = \emptyset$ contradicting the definition of F. If u is part of this cycle then the length of the cycle is at least 4. Let C be $\{u, \cdots, v, w, \cdots, x\}$ and construct C' as $\{u, w, \cdots, x\}$. Then $C' \cap F = \emptyset$ a contradiction. This proves the other direction. The number of vertices in the new instance of the graph is bounded by $(k + 1)n$ and this instance can be obtained in polynomial time from G. □

Corollary 1 Integer − WFVS *in tournaments can be solved in* $O((2.27)^k + (kn)^3)$ *time.*

Proof: Let T' be the graph obtained from T by applying Theorem 1. Then clearly T' is a tournament if T is. Now corollary follows from Lemma 1. □

If the weights are arbitrary reals, but at least 1, then the algorithm for unweighted tournament can not be directly applied. Here we give an algorithm which attains $O((2.4143)^k n^\omega)$ bound.

We will need the following observations to prove the correctness and the runtime of the algorithm.

Lemma 2 *[1] A tournament $T = (V, E)$ has a directed cycle if and only if it has a directed triangle.*

Let M be the adjacency matrix of an oriented directed graph T. Then T has a directed triangle if and only if for some i, j such that $1 \leq i < j \leq n$, $M^2[i, j] \geq 1$ and $M[j, i] = 1$. This can be determined in $O(n^\omega)$ time. If such a pair (i, j) exists, then there exists a k such that $M[i, k] = M[k, j] = 1$ which can also be determined in $O(n)$ time. Such a triple $\{i, j, k\}$ forms a triangle. Further, T has a directed cycle of length 4 if and only there exists a pair (i, j) such that $1 \leq i < j \leq n$, $M^2[i, j] \geq 1$ and $M^2[j, i] \geq 1$. If such a pair exists, then as before, the witness 4-cycle can also be found in $O(n)$ time. So we have

Lemma 3 *Let T be an oriented directed graph on n vertices. Then we can find a directed triangle or a directed cycle of length 4 in T, if it exists, in $O(n^\omega)$ time.*

We need the following lemma for our algorithm.

Lemma 4 *Let $T = (V, A)$ be a weighted tournament that does not contain a directed cycle of length 4. Then the minimum weight feedback vertex and arc set problems are solvable in T in $O(n^\omega)$ time.*

Proof: It is easy to see that if a tournament does not have directed cycle of length 4 then no pair of directed triangles in the tournament has a vertex in common. Hence the minimum weight feedback vertex or arc set is obtained by finding all triangles, and picking a minimum weight vertex/arc from each of them.

Finding all triangles in such a tournament can be done in $O(n^\omega)$ time as follows. First compute M^2, the square of the adjacency matrix of the tournament. Since the tournament can have at most $n/3$ triangles, there can be at most $n/3$ pairs (i, j) such that $1 \leq i < j \leq n$ and $M^2[i, j] \geq 1$ and $M[j, i] = 1$. For each such pair, the corresponding witness triangle can be found in $O(n)$ time. $\square$

We remark here that a tournament T has a directed cycle of length 4 if and only if it has a subgraph isomorphic to F_1 (see figure 2) and F_1 can be found in $O(n^{\omega})$ time.

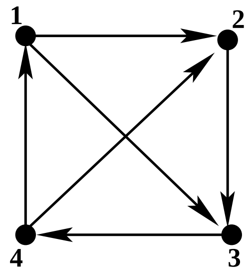

Fig. 2. F_1

Algorithm TFVS(T, k, π, F)(*T is a tournament, $k \geq 0$, π is a weight function on V, F is a set of vertices*)
(Returns 'true' and a minimal feedback vertex set of weight at most k, if exists and returns 'no' otherwise. F contains vertices of a partial feedback vertex set that are deleted from the original T. Initially the algorithm is called by TFVS$(T, k, \pi, \emptyset)$.)

Step 0: If T does not have a directed triangle and $k \geq 0$, then return 'true' and F and 'exit'.
Step 1: If $k = 0$ and T has a triangle, then answer 'no' and 'exit'.
Step 2: Find an induced subgraph on 4 vertices isomorphic to F_1 (as in figure 2) with vertex set, say $\{1, 2, 3, 4\}$ and the adjacencies be as in figure 2. If no such subgraph exists, then go to Step 4.
Step 3: If any of the following recursive calls results in true, then return 'true' and the corresponding F and 'exit', else return 'no' and 'exit'. In the following, T' is obtained by deleting the 'newly included' vertices in F.
(1) $TFVS(T', k - \pi(3), \pi, F \cup \{3\})$,
(2) $TFVS(T', k - \pi(4), \pi, F \cup \{4\})$,
(3) $TFVS(T', k - \pi(1) - \pi(2), \pi, F \cup \{1, 2\})$
Step 4: Find a minimum weight feedback vertex set S for the resultant tournament using Lemma 4 in polynomial time. If $\pi(S) > k$ then return 'no' and 'exit' else return 'true' and $F \cup S$ and 'exit'. Here $\pi(S) = \sum_{v \in S} \pi(v)$.

Correctness of Step 0 and 1 follow from Lemma 2. In Step 3, we branch on all possible minimal solutions of F_1. Step 4 follows from Lemma 4.

Since the weight of each vertex is at least 1, the time taken by the algorithm is bounded by the following recurrence:

$$T(n, k) \leq 2T(n - 1, k - 1) + T(n - 2, k - 2) + O(n^{\omega})$$

which solves to $O((2.4143)^k n^{\omega})$. So we have following theorem:

Theorem 2 *Given a tournament $T = (V, E)$, and a weight function $\pi : V \rightarrow$*

$\Re^+$, such that $\pi(v)$ is at least 1 for every $v \in V$, we can determine whether T has a feedback vertex set of weight at most k in $O((2.4143)^k n^\omega)$ time.

General-WFVS problem, where weights can be arbitrary positive reals, is not fixed parameter tractable unless $P = NP$. We show this by proving that it is NP-complete for some fixed constant k (in fact, for $k = 1$). Our reduction is from the NP-complete unweighted feedback vertex set problem [19] in tournaments. Let T be a tournament on n vertices where we are interested in finding a FVS of size k'. Define the weight function π to be $\pi(v) = 1/k'$ for all $v \in V$. Then the original tournament has a FVS of size k' if and only if the resulting weighted tournament has a FVS of weight 1. This implies that there cannot be a $f(k)n^{O(1)}$ or even an $n^{O(k)}$ time algorithm for General-WFVS problem unless $P = NP$. This result is true for general directed graph since unweighted feedback arc set problem is NP-complete for general directed graphs.

Theorem 3 *General-WFVS problem is not fixed parameter tractable in general directed graphs unless $P = NP$.*

3 Feedback Arc Set Problem in Tournaments

3.1 Feedback Arc Set Problem in Tournaments is FPT

We will first obtain a bound on the length of a shortest cycle in a graph with at most k arcs away from a tournament.

Lemma 5 *Let $G = (V, E)$ be a directed graph such that $|V| = n$ and $|E| \geq \binom{n}{2} - k$, for some non-negative integer k. Then either G is acyclic or has a directed cycle of length at most $c\sqrt{k}$ for some positive constant c.*

Proof: Assume G is not acyclic and choose c such that $c^2k - 3c\sqrt{k} \geq 2k$ ($c = 3$ suffices for $k \geq 2$). Note that the shortest directed cycle C of G is chordless; i.e. for all non-adjacent pairs of vertices u, v in C, there is no arc (u, v) or (v, u) since otherwise that arc between u and v will give rise to a shorter directed cycle. Suppose that the length l of the shortest directed cycle C in G is strictly greater than $c\sqrt{k}$. Then G misses $\binom{l}{2} - l = l(l-3)/2 > (c^2k - 3c\sqrt{k})/2 \geq k$ arcs. This is a contradiction since G has at least $\binom{n}{2} - k$ arcs. $\square$

Now we are ready to show the following theorem.

Theorem 4 *Given a tournament $T = (V, E)$, we can determine whether it has a feedback arc set of size at most k in $O((c\sqrt{k/e})^k n^\omega \lg n)$ time, where* *c is a positive constant. I.e. the feedback arc set problem is fixed parameter tractable in tournaments. (Here e is the base of natural logarithm.)*

Proof: We give an algorithm *TFES* which constructs a search tree for which each node has at most $c\sqrt{k}$ children. Each node in the tree is labelled with a set of vertices S that represents a partially constructed feedback arc set.

Algorithm TFES($T = (V, E)$, k, F) (* $k \geq 0$ *)
(Returns 'true' and a feedback arc set of size at most k, if exists and returns 'no' otherwise. F contains the arcs of a partial feedback arc set. Initially the algorithm is called by TFES($T, k, \emptyset$).)

Step 1: Find a shortest cycle C in T, if exists.
Step 2: If T is acyclic, then return 'true' and $\emptyset$ and 'exit'.
Step 3: If $k = 0$, then answer 'no' and 'exit'.
Step 4: If for some arc $e \in C$, TFES($T', k-1, F \cup \{e\}$) is true, where $T' = (V, E - e)$ then return 'true' and F (Note: Here F is actually $F \cup \{e\}$.) and 'exit', else answer 'no' and 'exit'.

If the algorithm exits at Step 2, then G can be made acyclic by not deleting any arc and hence its answer is correct (since $k \geq 0$). If it exits at Step 3, then G has a cycle and so it can't be made acyclic by deleting $k = 0$ arcs, and so its answer is correct. Finally the correctness of Step 4 follows from the fact that any feedback arc set must have one of these arcs of the cycle C, and the step is recursively checking for each arc e in the cycle whether $T - e$ has a feedback arc set of size at most $k - 1$.

To show that the algorithm takes the claimed bounds, observe that since k decreases at every recursive Step 4 (after an edge deletion), the recursion depth is at most k. Also the resulting directed graph after the i-th step of the recursion has at most i arcs deleted from a tournament. Hence Lemma 5 applies and so there is a cycle of length at most $c\sqrt{i}$ in the resulting graph after the i-th step. So the number of nodes in the search tree is $O(c^k\sqrt{k!})$. The shortest cycle in a directed graph can be found in $O(n^\omega \lg n)$ time [10]. Hence the claimed bound follows from Stirling's approximation. □

3.2 Improved Algorithms

The algorithms in this section are based on Lemma 6 which is observed independently by Gallai [8] and Grinberg et al. [9]. We give a proof here for completeness.

Lemma 6 (Reversal Lemma)

Let $G = (V, E)$ be a directed graph and F be a minimal feedback arc set (FAS) of G. Let G' be the graph formed from G by reversing the arcs of F in G. Then G' is acyclic.

Proof: Assume to the contrary that G' has a cycle C. Then C can not contain all the arcs of $E - F$, as that will contradict the fact that F is a FAS. Define the set $rev(F) = \{(u, v) \mid (v, u) \in F\}$. Let $C \cap rev(F) = \{f_1, f_2, \cdots, f_k\}$ and $e_i = rev(f_i)$. Then the set $\{e_1, e_2, \cdots, e_k\}$ is a set of arcs of G which are reversed and are part of C. Now since each $e_i \in F$, and F is minimal, there exists a cycle C_i in G such that $F \cap C_i = \{e_i\}$. Now consider the directed graph L induced by the arcs of $\{C, C_1, \cdots, C_k\} - F - rev(F)$. It is clear that L is a directed closed walk with all the arcs in the original graph G. In fact, if $\forall i,\ C_i \cap C = \emptyset$, then L is a simple cycle in G, such that $L \cap F = \emptyset$, contradicting the fact that F is a FAS. If L is not a simple cycle then we can extract a simple directed cycle from it not having any arcs of F, violating the definition of F. $\square$

Now we use Lemma 6 to give an improved algorithm for the feedback arc set problem in a tournament.

Algorithm TFAS(T,k, F)(*T is a tournament, $k \geq 0$, and F is a set of arcs.*)
(Returns 'true' and a minimal feedback arc set of size at most k, if exists and returns 'no' otherwise. F contains the arcs of a partial feedback arc set that are reversed from the original T. Initially the algorithm is called by TFAS($T, k, \emptyset$).)

Step 0: If T does not have a directed triangle and $k \geq 0$, then return 'true' and F.
Step 1: If $k = 0$ and T has a triangle, then answer 'no' and 'exit'.
Step 2: Find a triangle in T and let $\{a, b, c\}$ be the arcs of the triangle.
Step 2a: If $rev(a), rev(b)$ and $rev(c)$ are in F, then answer 'no' and 'exit'.
Step 2b: If $TFAS(T\backslash\{x\} \cup rev\{x\}, k-1, F \cup \{x\})$ is true for any arc x of the triangle such that $rev(x)$ is not in F, then return 'true' and F (Note: Here F is actually $F \cup \{x\}$.) and 'exit'. Otherwise return 'no' and 'exit'.

Theorem 5 *Given a tournament $T = (V, E)$ on n vertices, we can determine whether it has a feedback arc set of size at most k in $O(3^k n^\omega)$ time.*

Proof: First we will show that the algorithm $TFAS$ finds a minimal feedback arc set of size at most k if exists. Correctness of Step 0 and Step 1 follow from

Lemma 2. Step $2a$ answers correctly as by Reversal Lemma, the current F can not be extended to a minimal feedback arc set of G. In Step $2b$, we branch on each arc x of the triangle such that $rev(x) \notin F$, because if none of these arcs is picked in the feedback arc set of G, then this triangle will survive in G', obtained by reversing the arcs of F. But then by Reversal Lemma, this F is not minimal. So this proves the correctness of the algorithm.

The claimed time bound can easily be seen by observing that k decreases at every recursive Step $2b$ by 1. So the recursion depth is at most k. The branching factor at every recursion step is at most 3 and hence by Lemma 3, we have the desired time bound for the algorithm. $\square$

We further improve the bound using a better branching technique.

Algorithm BTFAS(T,k, F)(*T is a tournament, $k \geq 0$, F is a set of arcs*) (Returns 'true' and a minimal feedback arc set of size at most k, if exists and returns 'no' otherwise. F contains the arcs of a partial feedback arc set that are reversed from the original T. Initially the algorithm is called by BTFAS($T, k, \emptyset$).)

Step 0: If T does not have a directed triangle, then return 'true' and F.
Step 1: If $k = 0$ and T has a triangle, then answer 'no' and 'exit'.
Step 2: Find an induced subgraph on 4 vertices isomorphic to F_1 (as in figure 2), if exists, in T. Such a subgraph is simply a tournament on 4 vertices having at least two directed triangles. Let the vertex set of such an F_1 be $\{1, 2, 3, 4\}$ and the adjacencies be as in figure 2 (in particular $(1, 2)$ is the only arc not part of any directed triangle). If no such subgraph exists in T, then go to Step 6.
Step 3: Let $\{a, b, c\}$ be the arcs of a triangle in F_1, such that there exists an arc $x \in \{a, b, c\}$ for which $rev(x) \in F$. If there is no such triangle in F_1, then go to Step 4.
Step 3a: If $rev(a), rev(b)$ and $rev(c)$ are in F, then answer 'no' and 'exit'.
Step 3b: If $BTFAS(T \backslash \{x\} \cup rev(x), k-1, F \cup \{x\})$ is true for any arc x of the triangle such that $rev(x)$ is not in F, then return 'true' and F (Note: Here F is actually $F \cup \{x\}$.) and 'exit'; else answer 'no' and 'exit'.
Step 4: If $rev((1, 2)) \notin F$ then if any of the following recursive calls returns true, then return 'true' and the corresponding F and 'exit', and answer 'no' and 'exit' otherwise.

In the following, T' is obtained from T by reversing the 'newly included' arcs of F.

(1) $BTFAS(T', k-1, F \cup \{(3, 4)\})$,
(2) $BTFAS(T', k-2, F \cup \{(4, 1), (4, 2)\})$,
(3) $BTFAS(T', k-2, F \cup \{(4, 1), (2, 3)\})$,
(4) $BTFAS(T', k-2, F \cup \{(1, 3), (2, 3)\})$,

(5) $BTFASF(T', k-3, F \cup \{(1,2),(1,3),(4,2)\})$

Step 5: If $rev((1,2)) \in F$, then if any of the first 4 recursive calls enumerated in Step 4 returns true, then return 'true' and the corresponding F and 'exit', and answer 'no' and 'exit' otherwise.

Step 6: Find a minimum feedback arc set S of a resultant tournament using Lemma 4 in polynomial time. If $|S| > k$ then return 'no' and 'exit' else return 'true' and $F \cup S$ and 'exit'.

In the above algorithm at every step, we first find a graph isomorphic to F_1, and then if there exists a directed triangle in F_1 with all its arcs included in the partial feedback arc set (F) obtained so far, then we apply Lemma 6 and answer 'no'. Otherwise we branch on all the arcs x of the triangle such that $rev(x) \notin F$ as by Lemma 6 at least one such arc must be part of F.

If none of the arcs of F_1 is part of F, then we branch on all possible minimal feedback arc sets of F_1. The only remaining case is when all the arcs x appearing in some triangle in F_1 are not in F but $rev((1,2)) \in F$. In this case, Lemma 6 implies that item 5 of *Step* 2*b* is not applicable (because the set $\{(1,3),(4,2)\}$ is not a minimal FAS of F_1). So when we reach *Step* 6 of the above algorithm, all the induced subgraphs on 4 vertices have at most one triangle. And the problem now can be solved in polynomial time by Lemma 4 .

Thus, we get the following recurrence for the time complexity of the algorithm:

$$T(n,k) \leq \max \begin{cases} 2T(n,k-1) + O(n^\omega) \text{ or} \\ T(n,k-1) + 3T(n,k-2) + T(n,k-3) + O(n^\omega) \end{cases}$$

The above recurrences solve to $O((2.415)^k n^\omega)$. So we get the following theorem.

Theorem 6 *Given a tournament $T = (V,E)$, we can determine whether it has a feedback arc set of size at most k in $O((2.415)^k n^\omega)$ time.*

We remark that the above algorithm can also be applied for weighted feedback arc set problem in a tournament where the weight of every arc is at least 1.

Theorem 7 *Given a tournament $T = (V,E)$, and a weight function $\pi : E \rightarrow \Re^+$, such that $\pi(e)$ is at least 1 for every $e \in E$, we can determine whether T has a feedback arc set of weight at most k in $O((2.415)^k n^\omega)$ time.*

4 Feedback Arc Set Problem in Dense Directed Graphs

In this section, we show that the feedback arc set problem is fixed parameter tractable for directed graphs which are at most $n^{1+o(1)}$ arcs away from a tournament. We need the following lemma to show the desired result. The girth of a graph is defined as the length of the shortest cycle in the graph. A directed graph is called strongly connected if there exists a directed path between every pair of vertices.

Lemma 7 *[2] Let $G = (V, E)$ be a strongly connected directed graph with n vertices, m arcs and let $l \geq 2$. Then if $m \geq \frac{n^2+(3-2l)n+(l^2-l)}{2}$, the girth of the graph ($g(G)$) is bounded by l.*

Corollary 2 *Let G be a strong directed graph with n vertices and $m \geq \binom{n}{2} - \frac{n(g-2)}{2}$ where $3 \leq g \leq n-6$. Then $g(G) \leq g$.*

Proof: Lemma 7 implies that if a strong directed graph has at least $\binom{n}{2} - \frac{n(g-2)}{2}$ arcs, then its girth is bounded by g. This is because

$$\binom{n}{2} - \frac{n^2+(3-2g)n+(g^2-g)}{2} = \frac{2n(g-2)+g-g^2}{2}$$

and

$$\frac{2n(g-2)+g-g^2}{2} \geq n(g-2) - \frac{g^2}{2} \geq \frac{n(g-2)}{2} \qquad \text{whenever } n \geq \tfrac{g^2}{g-2} .$$

To show $n \geq \frac{g^2}{g-2}$, it suffices to show

$$\begin{aligned} n \geq & \frac{g^2}{g-2} \\ = & \frac{g^2-4}{g-2} + \frac{4}{g-2} \\ = & g+2+\frac{4}{g-2}. \end{aligned}$$

We know $g + 2 + \frac{4}{g-2} \leq g + 6 \leq n$, which completes the claim. □

Theorem 8 *Let G be a directed graph with n vertices and $m \geq \binom{n}{2} - n^{1+o(1)}$ arcs. Then the feedback arc set (FAS) problem is fixed parameter tractable for G.*

Proof: For the feedback arc set problem, we can assume without loss of generality, that the given directed graph is a strongly connected directed graph. (Otherwise, try values up to k in each strongly connected subgraph and take the minimum.)

We find the shortest cycle in G and then by applying Lemma 6, we branch on each arc by reversing the arc. This way we don't delete any arc and hence at every recursive step Corollary 2 ensures a cycle of length at most $n^{o(1)}$. So we have an algorithm for feedback arc set problem in G which takes $O((n^{o(1)})^k n^{O(1)})$ time. Cai and Judes [3] have observed that $O((n^{o(1)})^k)$ algorithm can be simulated by an algorithm of time $f(k)n^{O(1)}$, where f is some function of k, for every fixed n and k. Hence it follows that the feedback arc set problem is fixed parameter tractable for G. $\Box$

Note that the proof does not carry over to the FVS problem on dense directed graphs. This is because, we may not obtain a dense directed graph after deleting a vertex from a dense directed graph.

5 Parametric Duals

The parametric dual of a parameterized problem with parameter k is the same problem with k replaced by 'all but k' ([11,12]). For example, the parametric dual of the k-vertex cover is the $(n-k)$- vertex cover or equivalently k-independent set problem.

In this section, we show that the parametric dual problems of the directed feedback set problems are themselves some natural optimization problems and their parameterized versions are fixed parameter tractable in oriented directed graphs.

5.1 Parametric Dual of Directed Feedback Vertex Set

Parameteric dual of directed feedback vertex set is : given a directed graph on n vertices, are there at most $n-k$ vertices whose removal makes the graph acyclic. Or equivalently, is there a set of at least k vertices that induces an acyclic directed graph? We call this the $maxv - acyclic$ subgraph problem.

Given a tournament T, we can find a subset S of vertices such that $|S| \geq \lfloor \lg n \rfloor$ and the induced subtournament of T on S is acyclic (transitive). We *repeatedly* include the vertex with the smallest indegree in the given tournament into S and remove it and its neighbours from the tournament. It is also clear that the induced subtournament on S is acyclic. For, if we order the vertices by the order in which they are included in S, then the arcs go only from smaller vertices to bigger vertices.

To show that $|S| \geq \lfloor \lg n \rfloor$, it suffices to show that the 'repeat' loop will execute for at least $\lfloor \lg n \rfloor$ steps. This follows because in any tournament there is a vertex with indegree at most $(n-1)/2$. Thus, after one step of the loop at most $(n+1)/2$ vertices are deleted.

Any oriented directed graph can be completed to a tournament by adding the missing arcs (with arbitrary directions). Hence every oriented directed graph G on n vertices and m arcs has at least $\lfloor \lg n \rfloor$ vertices that induce an acyclic subgraph.

Let the oriented directed graph G be given as an adjacency list where associated with every vertex x is a list of vertices y such that y is an inneighbour of x. It is easy to implement each step of the 'repeat' loop in $O(m)$ time (We can have a bitvector for the list of vertices to be deleted and scan through the adjacency list and remove those vertices.). By exiting the loop when $|S| = \lfloor \lg n \rfloor$, we get the following lemma.

Lemma 8 *Let G be an oriented directed graph with n vertices and m arcs. Then there exists a subset of $\lfloor \lg n \rfloor$ vertices which induces an acyclic subgraph and it can be found in $O(\min\{m \lg n, n^2\})$ time.*

Now we can design a fixed parameter tractable algorithm for the parameterized $maxv - acyclic$ subgraph problem as follows. If $k \leq \lfloor \lg n \rfloor$, then return the acyclic subgraph obtained in Lemma 8, otherwise $n \leq 2^k$ and then we check all k sized subsets of the vertex set to see whether the subset induces an acyclic subgraph. If any one of them does, then we return the acyclic subgraph, otherwise answer 'no'. Since $\binom{n}{k} \leq \binom{2^k}{k} \leq (e2^k/k)^k$, we have the following theorem. (Here e is the base of natural logarithm.)

Theorem 9 *Given an oriented directed graph G and an integer k, we can determine whether or not G has at least k vertices that induce an acyclic subgraph in time $O((e2^k/k)^k k^2 + min(\{m \lg n\}, \{n^2\}))$; i.e $maxv - acyclic$ subgraph problem is fixed parameter tractable.*

The fixed parameter tractable algorithm for the $maxv - acyclic$ subgraph problem follows from the easy observation that there is a "guarantee" (lower bound) of $\lfloor \lg n \rfloor$ for the solution size. In such situations, it is natural to parameterize above the guarantee [12] and so a natural question to ask is whether a given directed graph has a set of at least $\lfloor \lg n \rfloor + k$ vertices that induces an acyclic subgraph. The parameterized complexity of this question is open. However it turns out that $maxv - acyclic$ subgraph problem is $W[1]$-hard in general directed graphs.

Theorem 10 *It is $W[1]$-hard to determine whether a given directed graph has k vertices that induce an acyclic subgraph; i.e. $maxv - acyclic$ subgraph problem is $W[1]$-hard in directed graphs.*

Proof: We reduce the k-independent set problem in undirected graphs to the given problem. Given an undirected graph $G = (V, E)$, an instance of the independent set problem we construct $D = (V, E^{'})$, an instance of $maxv - acyclic$ subgraph problem in directed graph by adding both arcs $u \rightarrow v$ and $v \rightarrow u$ for every (u, v) in E. If G has an independent set of size k, then those corresponding vertices of D form an acyclic subgraph. Conversely, if D has an acyclic subgraph on k vertices, then those k vertices must form an independent set in D as if there is an arc between a pair of vertices in D, then there actually is a directed cycle (of length 2) between them. □

5.2 Parametric Dual of Directed Feedback Arc Set

Parametric dual of directed feedback arc set is : given a directed graph on n vertices and m arcs, are there at most $m - k$ arcs whose removal makes the graph acyclic. Or equivalently, is there a set of at least k arcs that induces an acyclic directed graph? We call this the $maxe-acyclic$ subgraph problem and show that it is fixed parameter tractable. It follows from the following easy lemma.

Lemma 9 *Given a directed graph G on n vertices and m arcs, there always exists a set of at least $\lceil m/2 \rceil$ arcs that form an acyclic directed graph. Such a set of arcs can be found in $O(m)$ time.*

Proof: Order the vertices of the directed graph G arbitrarily. If m is the number of arcs in the graph, then at least $\lceil m/2 \rceil$ of these arcs go in one direction (all from a smaller vertex to a higher vertex or vice versa). These arcs form an acyclic directed graph. □

Theorem 11 *Given a directed graph G on n vertices and m arcs and an integer parameter k, we can determine whether or not G has at least k arcs that form an acyclic subgraph in time $O(4^k k + m)$, i.e $maxe-acyclic$ subgraph problem is FPT in directed graphs.*

Proof: If $k \leq m/2$, then return the acyclic subgraph obtained in Lemma 9, otherwise $m \leq 2k$, and then check all k subsets of the arc set of the graph. If any of these k subsets of arcs induces an acyclic subgraph, then return the acyclic subgraph, and answer 'no' otherwise.

Since $\binom{m}{k} \le 2^m \le 2^{2k}$, and we can check in $O(k)$ time if k arcs forms an acyclic graph, we have the desired running time. $\square$

Just as in the parametric dual of the feedback vertex set problem, a natural parameterized question here is whether the given directed graph has a set of at least $\lceil m/2 \rceil + k$ arcs that form an acyclic subgraph. This question is open for general directed graphs. In fact, $\lceil m/2 \rceil$ is a tight lower bound for the solution size in directed graphs. This bound is realized, for example, in a directed graph obtained by taking an undirected path on n vertices (and n edges) and replacing every edge by a pair of directed arcs one in each direction.

However, this bound of $\lceil m/2 \rceil$ is not tight for oriented directed graphs.

We prove that there exists an acyclic subgraph on $\frac{m}{2} + 1/2\lceil (n-c)/2 \rceil$ arcs in any oriented directed graph and then use this to give fixed parameter algorithm for the question 'whether the given oriented directed graph has a set of at least $\frac{m}{2} + k$ arcs that forms an acyclic subgraph'.

We mimic the proof of the following lemma proved in [14].

Lemma 10 *[14] If G is a simple undirected graph (without parallel edges and self loops) with m edges, n vertices and c components, then the maximum number of edges in a bipartite subgraph of G is at least $\frac{m}{2} + \frac{1}{2}\lceil (n-c)/2 \rceil$. Such a bipartite graph can be found in $O(n^3)$ time.*

If we just apply Lemma 10 on the underlying undirected graph then at least half the arcs of the bipartite subgraph returned by the Lemma 10 are in one direction and that gives us a lower bound of $\frac{m}{4} + \frac{1}{4}\lceil (n-c)/2 \rceil$ on the size of maximum acyclic subgraph of G. However, by modifying the proof of Lemma 10, we get a bound of $\frac{m}{2} + \frac{1}{2}\lceil (n-c)/2 \rceil$. We will use $a(G)$ to denote the size of a maximum acyclic subgraph of G.

Lemma 11 *Any oriented directed graph $G = (V, E)$ with m arcs and n vertices, with the underlying undirected graph having c components, has an acyclic subgraph with at least $\frac{m}{2} + 1/2\lceil (n-c)/2 \rceil$ arcs, i.e. $a(G) \ge \frac{m}{2} + 1/2\lceil (n-c)/2 \rceil$ and such a subgraph can be found in $O(n^3)$ time.*

Proof: Without loss of generality assume that the underlying undirected graph is connected, otherwise we will apply this lemma on each component to get the result. The proof is along the lines of the proof of Lemma 10. We give the proof for completion.

The proof is by induction on the number of vertices. The lemma is clearly true for oriented directed graphs on 1 or 2 vertices. At the induction step, there are three cases.

Case 1: The underlying undirected graph has a cut vertex x.

In this case, we apply induction on each of the connected components of the underlying undirected graph of $G - x$ including x in each component. Let the connected components of the underlying undirected graph of $G - x$ be $\{C_1, C_2, ..., C_k\}$ and let G_i denote the induced subgraph on $C_i \cup \{x\}$. Then we have:

$$a(G) \geq \sum_{i=1}^{k} a(G_i).$$

Let $E(G_i)$ denote the edge set of G_i and let m_i and n_i denote the cardinality of edges and vertices of G_i respectively. Observe that $\sum_i m_i = m$ and $\sum_i (n_i - 1) = n - 1$. Then by applying induction hypothesis on G_i, we get

$$\begin{aligned} a(G) &\geq \sum_{i=1}^{k} \frac{m_i}{2} + \frac{1}{2}\lceil (n_i - 1)/2 \rceil \\ &\geq \frac{m}{2} + \frac{1}{2}\lceil (n-1)/2 \rceil \\ &\quad \text{(Since } \lceil x \rceil + \lceil y \rceil \geq \lceil x + y \rceil \text{, for any two rationals } x \text{ and } y.\text{)} \end{aligned}$$

Case 2: The underlying undirected graph has no cut vertex and the oriented directed graph G has a vertex x whose indegree and outdegree are not the same.

In this case we apply induction on $G - x$ (whose underlying undirected graph will be clearly connected), and also include all arcs coming into x or all arcs going out of x whichever set is larger, into the acyclic subgraph to get an acyclic subgraph in the resulting directed graph. So we get:

$$\begin{aligned} a(G) &\geq a(G - x) + \frac{d_G(x) + 1}{2} \\ &\geq \frac{m - d_G(x)}{2} + \frac{1}{2}\lceil (n-2)/2 \rceil + \frac{d_G(x) + 1}{2} \\ &\geq \frac{m}{2} + \frac{1}{2}\lceil (n-1)/2 \rceil. \end{aligned}$$

Here $d_G(x)$ represents the number of neighbours (both inneighbour and outneighbour) of x in G.

Case 3: Every vertex has even degree and the underlying undirected graph has no cut vertex.

This implies that there exists a pair of adjacent vertices u and v such that the underlying undirected graph of $G - \{u, v\}$ is connected (for a proof see [14]). Apply induction on $G - \{u, v\}$ and pick all outgoing arcs from u and v. This implies that

$$
\begin{aligned}
a(G) &\geq a(G-u-v) + \frac{d_G(u)+d_G(v)}{2} \\
&\geq \frac{m-(d_G(u)+d_G(v)-1)}{2} + \frac{1}{2}\lceil (n-3)/2 \rceil + \frac{d_G(u)+d_G(v)}{2} \\
&= \frac{m}{2} + \frac{1}{2}\lceil (n-3)/2 \rceil + \frac{1}{2} \\
&\geq \frac{m}{2} + \frac{1}{2}\lceil (n-1)/2 \rceil.
\end{aligned}
$$

It is easy to verify that the resulting set of arcs forms an acyclic subgraph, and can be found in the claimed bound. □

Theorem 12 *Let G be an oriented directed graph on n vertex and m arcs. Let c be the number of components in the underlying undirected graph. Then given an integer k, we can determine whether or not G has at least $\frac{m}{2}+k$ arcs which forms an acyclic subgraph in time $O(c2^{O(k^2)}k^2+m+n^3)$.*

Proof: First, find all the c components of the underlying undirected graph corresponding to G. If $k \leq 1/2\lceil (n-c)/2 \rceil$, then G has an acyclic subgraph with at least $\frac{m}{2}+k$ arcs, else $k > 1/2\lceil (n-c)/2 \rceil \geq (n-c)/4-1$ or $n \leq 4k+4+c$. Thus n_i, the number of vertices in the i-th component is at most $n-(c-1) \leq 4k+5$. Hence the number of arcs in each of the components is $O(k^2)$. By trying all subsets of the arcs in the component, we can find m_i, the maximum number of arcs of the i-th component that form an acyclic subgraph, for any i, in $O(2^{O(k^2)}k^2)$ time. If $\sum_{i=1}^{c} m_i \geq \lceil m/2 \rceil + k$ then G has an acyclic subgraph with at least $\frac{m}{2}+k$ arcs, else G does not have an acyclic subgraph with at least $\frac{m}{2}+k$ arcs. □

6 Conclusions

In this paper, we have obtained efficient algorithms for parameterized feedback arc and vertex set problem on weighted tournaments. For the feedback arc set problem, the complexity of the algorithms in unweighted and weighted (with weights at least 1) versions are the same while this is not the case for the feedback vertex set problem.

The best known algorithm for FVS in unweighted tournaments is through the hitting set algorithm. It would be interesting to see whether the unweighted feedback vertex set problem on tournaments has some special structure that can be utilized to develop an algorithm better than that of the 3-hitting set problem.

We have also given FPT algorithms for the parametric duals of directed feedback vertex and arc set problems in oriented directed graphs and directed graphs respectively. Dual of directed feedback vertex set problem in directed graphs is shown to be $W[1]$-hard. In line with parameterizing above the guaranteed values, the parameterized complexity of the following questions are also interesting.

- Given an oriented directed graph on n vertices, does it have a subset of at least $\lfloor \lg n \rfloor + k$ vertices that induces an acyclic subgraph?
- Given an oriented directed graph on n vertices and m arcs, does it have a subset of at least $m/2 + 1/2(\lceil n - 1/2 \rceil) + k$ arcs that induces an acyclic subgraph ?
- Given a directed graph on n vertices and m arcs, does it have a subset of at least $m/2 + k$ arcs that induces an acyclic subgraph ?

7 Acknowledgement

We thank P. Paulraja, S. Srinivasa Rao for fruitful discussions and comments on earlier versions of this paper and the two anonymous referees for several useful comments.

References

[1] J. Bang-Jensen and G. Gutin, 'Digraphs Theory, Algorithms and Applications', (2001) Springer-Verlag.

[2] J. C. Bermond, A. Germa, M. C.Heydemann and D. Sotteau, 'Girth in Digraphs', *Journal of Graph Theory*, **4** (3) (1980) 337-341.

[3] L. Cai and D. Juedes, 'On the Existence of Subexponential Parameterized Algorithms', *Journal of Computer and System Sciences*, **67** (4) (2003) 789-807.

[4] R. Downey and M. R. Fellows, 'Parameterized Complexity', (1999) Springer-Verlag.

[5] C. Dwork, R. Kumar, M. Naor and D. Sivakumar, 'Rank Aggregation Revisited', Manuscript available at: *www.cs.northwestern.edu/ ∼ kao/cs395-ecommerce/reading_document_ranking/ecom.document_ranking.dwork.pdf.*

[6] G. Even, J. (Seffi) Naor, B. Schieber, M. Sudan, 'Approximating Minimum Feedback Sets and Multicuts in Directed Graphs', *Algorithmica* , **20** (1998) 151-174.

[7] M. Fellows, M. Hallett, C. Korostensky, U. Stege, 'Analogs and Duals of the MAST Problem for Sequences and Trees', *Journal of Algorithms*, **49** (1) (2003) 192-216.

[8] T. Gallai, 'On Directed Paths and Circuits', *in Theory of Graphs, Proceedings Colloquium Tihnay, P Erdős, G. Katona Eds.*, (1968) 115-118.

[9] E. Ya. Grinberg and Ya. Ya. Dambit, 'Nekotorye Svoistva Grafov Soderzhashchikh Kontury' [Russian: Some Properties of Directed Graphs With Circuits], *Latviiskii Mathematicheskii Ezhegodnile*, **2** (1966) 65-70.

[10] A. Itai and M. Rodeh, 'Finding a Minimum Circuit in a Graph', *SIAM Journal on Computing*, **7** (4) (1978) 413-423.

[11] S. Khot and V. Raman, 'Parameterized Complexity of Finding Subgraphs with Hereditary Properties', *Theoretical Computer Science*, **289** (2002) 997-1008.

[12] M. Mahajan and V. Raman, 'Parameterizing above Guaranteed Values: MaxSat and MaxCut', *Journal of Algorithms*, **31** (1999) 335-354.

[13] R. Niedermeier and P. Rossmanith, 'An efficient Fixed Parameter Algorithm for 3-Hitting Set', *Journal of Discrete Algorithms*, **1** (1) (2003) 89-102.

[14] S. Poljak and D. Turzik, 'A Polynomial Algorithm for Constructing a Large Bipartite Subgraph, with an Application to a Satisfiability Problem',*Canad. J. Math*, **34** (3) (1982) 519-524.

[15] V. Raman, S. Saurabh and C. R. Subramanian, 'Faster Fixed Parameter Tractable Algorithms for Undirected Feedback Vertex Set', *in the Proceedings of the 13th International Symposium on Algorithms and Computation (ISAAC 2002)*, Lecture Notes in Computer Science, **2518** (2002) 241-248 Springer-Verlag.

[16] V. Raman, S. Saurabh and C. R. Subramanian, 'Faster Algorithms for Feedback Vertex Set', To appear *in the Proceedings of the 2nd Brazilian Symposium on Graphs, Algorithms and Combinatorics (GRACO 2005).*

[17] V. Raman and S. Saurabh, 'Parameterized Complexity of Directed Feedback Set Problems in Tournaments', *in the Proceedings of Workshop on Data Structures and Algorithms (WADS 2003)*, Lecture Notes in Computer Science, **2748** (2003) 484-492 Springer-Verlag.

[18] V. Raman and S. Saurabh, 'Improved Parameterized Algorithms for Feedback Set Problems in Weighted Tournaments', *in the Proceedings of International Workshop on Parameterized and Exact Computation (IWPEC 2004)*, Lecture Notes in Computer Science, **3162** (2004) 260-270 Springer-Verlag.

[19] E. Speckenmeyer, 'On Feedback Problems in Digraphs', *in Proceedings of the 15th International Workshop WG'89*, Lecture Notes in Computer Science, **411** (1989) 218-231 Springer-Verlag.