Parameterized Complexity of the Anchored k-Core Problem for Directed Graphs*

Rajesh Chitnis[1], **Fedor V. Fomin**[2], **and Petr A. Golovach**[2]

1 Department of Computer Science, University of Maryland at College Park, USA
rchitnis@cs.umd.edu
2 Department of Informatics, University of Bergen, Bergen, Norway
{fedor.fomin,petr.golovach}@ii.uib.no

Abstract

We consider the DIRECTED ANCHORED k-CORE problem, where the task is for a given directed graph G and integers b, k and p, to find an induced subgraph H with at least p vertices (the core) such that all but at most b vertices (the anchors) of H have in-degree at least k. For undirected graphs, this problem was introduced by Bhawalkar, Kleinberg, Lewi, Roughgarden, and Sharma [ICALP 2012]. We undertake a systematic analysis of the computational complexity of DIRECTED ANCHORED k-CORE and show that:

- The decision version of the problem is NP-complete for every $k \geq 1$ even if the input graph is restricted to be a planar directed acyclic graph of maximum degree at most $k+2$.
- The problem is fixed parameter tractable (FPT) parameterized by the size of the core p for $k = 1$, and W[1]-hard for $k \geq 2$.
- When the maximum degree of the graph is at most Δ, the problem is FPT parameterized by $p + \Delta$ if $k \geq \frac{\Delta}{2}$.

1998 ACM Subject Classification F.2.2, G.2.1, G.2.2

Keywords and phrases Parameterized complexity, directed graphs, anchored k-core

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2013.79

1 Introduction

Degree-constrained subgraph problems have been extensively studied in theoretical computer science. One can describe degree constrained subgraph problems in the following general setting: given a (un)directed graph G, find a maximum/minimum sized (induced, connected) subgraph H subject to some condition $\mathcal{C}$ imposed on the degrees of vertices. For example, INDEPENDENT SET or (INDUCED) MATCHING can be seen as problems within this framework. In this paper, we study an interesting variant of the degree-constrained subgraph problem where we have to find a large subgraph with all vertices satisfying constrains except a small set of anchors vertices. While such type of problems arise naturally in different settings, in particular in social sciences, adding of anchors can bring to non-trivial computational challenges.

* The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP/2007-2013)/ERC Grant Agreement n. 267959 and from NSF CAREER award 1053605, NSF grant CCF-1161626, ONR YIP award N000141110662, DARPA/AFOSR grant FA9550-12-1-0423, a University of Maryland Research and Scholarship Award (RASA) and a Summer International Research Fellowship from the University of Maryland.

More precisely, the k-core of a directed graph G is defined as the largest subgraph H such that $\deg_H^-(v) \geq k$ for every $v \in V(H)$. This notion was introduced by Seidman [17] and is a well-known concept in the theory of social networks. It has also been studied in various social sciences literature [8, 9]. It is easy to see that we can find the k-core of a given directed graph in polynomial time by the following procedure: iteratively remove any vertex that has in-degree less than k. However, one might not want to strictly enforce the condition of in-degree being at least k for every vertex. In particular, we allow for a small number of special vertices (called anchors) which can have arbitrary in-degrees, but their purpose in the k-core is to augment the in-degrees of the non-anchored vertices. Bhawalkar et al. [2] introduced the Anchored k-Core problem for (undirected) graphs. In the Anchored k-Core problem the input is an undirected graph $G = (V, E)$ and integers b, k, and the task is to find an induced subgraph H of maximum size with all vertices but at most b (which are anchored) to be of degree at least k. In this work we extend the notion of anchored k-core to directed graphs and define the parameterized version of the problem formally:

Directed Anchored k-Core (Dir-AKC)
Input: A directed graph $G = (V, E)$ and integers b, k, p.
Parameter 1: b.
Parameter 2: k.
Parameter 3: p.
Question: Do there exist sets of vertices $A \subseteq U \subseteq V(G)$ such that $|A| \leq b$, $|U| \geq p$, and every $v \in U \setminus A$ satisfies $d^-_{G[U]}(v) \geq k$?

We will refer to the set A as the set of *anchors* and to the graph $H = G[U]$ as the *anchored k-core*. Note that the undirected version of Anchored k-Core problem can be modeled by the directed version: simply replace each edge $\{u, v\}$ by arcs (u, v) and (v, u). Keeping the parameters b, k, p unchanged it is now easy to see that the two instances are equivalent.

Connection to Preventing Unraveling in Social Networks: Social networks are generally represented by making use of undirected or directed graphs, where the edge set represents the relationship between individuals in the network. The undirected graph model works fine for some networks, say Facebook, but the nature of interaction on some social networks such as Twitter is asymmetrical: the fact that user A follows user B does not imply that user B also follows A. In this case, it is more appropriate to model interactions in the network by *directed* graphs. We add a directed edge (u, v) if v follows u. We can consider a model of *user management* where there is a threshold value k, such that each individual with less than k people to follow (or equivalently whose in-degree is less than k) drops out of the network. This process can be contagious, and may affect even those individuals who initially were linked to more than k people. An extreme example of this was given by Schelling (see page 17 of [15]): consider a directed path on n vertices and let $k = 1$. The left-endpoint has in-degree zero, it drops out and now the in-degree of its only out-neighbor in the path becomes zero and it drops out as well. It is not hard to see that this way the whole network eventually drops out as the result of a *cascade of iterated withdrawals*, i.e., the 1-core of this graph is the empty set. The unraveling process described above in Schelling's example of a directed path can be highly undesirable in many scenarios. One can attempt to prevent this unraveling by introducing a few special vertices (called anchors) by "buying" them with extra incentives.

Parameterized Complexity: We are mainly interested in the parameterized complexity of Anchored k-Core. For the general background, we refer to the books by Downey and Fellows [10], Flum and Grohe [12] and Niedermeier [14]. Parameterized complexity is basically a two dimensional framework for studying the computational complexity of a problem. One dimension is the input size n and another one is a parameter k. A problem is said to be *fixed parameter tractable* (or FPT) if it can be solved in time $f(k) \cdot n^{O(1)}$ for some function f. A problem is said to be in XP, if it can be solved in time $O(n^{f(k)})$ for some function f. The W-hierarchy is a collection of computational complexity classes: we omit the technical definitions here. The following relation is known amongst the classes in the W-hierarchy: FPT $=$ W[0] $\subseteq$ W[1] $\subseteq$ W[2] $\subseteq \ldots$. It is widely believed that FPT $\neq$ W[1], and hence if a problem is hard for the class W[i] (for any $i \geq 1$) then it is considered to be fixed-parameter intractable.

Previous Results: Bhawalkar et al. [2] initiated the algorithmic study of Anchored k-Core on undirected graphs. In particular, they obtained the following dichotomy result: the decision version of the problem is solvable in polynomial time for $k \leq 2$ and is NP-complete for all $k \geq 3$. Moreover, for $k \geq 3$ the problem remains NP-complete even on planar graphs [6]. This motivates the study of the problem for $k \geq 3$ from the viewpoint of parameterized complexity. Unfortunately, the problem is W[2]-hard parameterized by b [2] and W[1]-hard parameterized by p even for $k = 3$ [6].

Our Results: In this paper we initiate the study of Anchored k-Core on directed graph and provide a new insight into the computational complexity of the problem. We obtain the following results.

- The decision version of Dir-AKC is NP-complete for every $k \geq 1$ even if the input graph is restricted to be a planar directed acyclic graph (DAG) of maximum degree at most $k + 2$. Thus the directed version is in some sense strictly harder than the undirected version which is known be in P if $k \leq 2$, and NP-complete if $k \geq 3$ [2]. These results are proven in Section 2.
- The NP-hardness result for Dir-AKC motivates us to make a more refined analysis of the Dir-AKC problem via the paradigm of parameterized complexity. We obtain (Section 3) the following dichotomy result: Dir-AKC is FPT parameterized by p if $k = 1$, and W[1]-hard if $k \geq 2$.

This fixed-parameter intractability result parameterized by p forces us to consider the complexity on special classes of graphs such as bounded-degree directed graphs or directed acyclic graphs.

- In Section 4, for graphs of degree upper bounded by Δ, we show that the Dir-AKC problem is FPT parameterized by $p + \Delta$ if $k \geq \frac{\Delta}{2}$. In particular, it implies that Dir-AKC is FPT parameterized by p for directed graphs of maximum degree at most four.
- We complement tractability results by showing in Section 5 that if $k < \frac{\Delta}{2}$ and $\Delta \geq 3$, then Dir-AKC is W[2]-hard when parameterized by the number of anchors b even for DAGs. On the other hand, the problem is FPT when parameterized by $\Delta + p$ for DAGs of maximum degree at most Δ. Note that we can always assume that $b \leq p$, and hence any FPT result with parameter b implies FPT result with parameter p as well. On the other side, any hardness result with respect to p implies the same hardness with respect to b.

Due to space limitations, some proofs are omitted here. They can be found in [5].

2 Preliminaries

We consider finite directed and undirected graphs without loops or multiple arcs. The vertex set of a (directed) graph G is denoted by $V(G)$ and its edge set (arc set for a directed graph) by $E(G)$. The subgraph of G induced by a subset $U \subseteq V(G)$ is denoted by $G[U]$. For $U \subset V(G)$ by $G - U$ we denote the graph $G[V(G) \setminus U]$. For a directed graph G, we denote by G^* the undirected graph with the same set of vertices such that $\{u, v\} \in E(G^*)$ if and only if $(u, v) \in E(G)$. We say that G^* is the *underlying* graph of G.

Let G be a directed graph. For a vertex $v \in V(G)$, we say that u is an *in-neighbor* of v if $(u, v) \in E(G)$. The set of all in-neighbors of v is denoted by $N_G^-(v)$. The *in-degree* $d_G^-(v) = |N_G^-(v)|$. Respectively, u is an *out-neighbor* of v if $(v, u) \in E(G)$, the set of all out-neighbors of v is denoted by $N_G^+(v)$, and the *out-degree* $d_G^+(v) = |N_G^+(v)|$. The *degree* $d_G(v)$ of a vertex v is the sum $d_G^-(v) + d_G^+$, and the *maximum degree* of G is $\Delta(G) = \max_{v \in V(G)} d_G(v)$. A vertex v of $d_G^-(v) = 0$ is called a *source*, and if $d_G^+(v) = 0$, then v is a *sink*. Observe that isolated vertices are sources and sinks simultaneously.

Let G be a directed graph. For $u, v \in V(G)$, it is said that v can be *reached* (or *is reachable*) from u if there is a directed $u \to v$ path in G. Respectively, a vertex v can be reached from a set $U \subseteq V(G)$ if v can be reached from some vertex $u \in U$. Notice that each vertex is reachable from itself. We denote by $R_G^+(u)$ ($R_G^+(U)$ respectively) the set of vertices that can be reached from a vertex u (a set $U \subseteq V(G)$ respectively). Let $R_G^-(u)$ denote the set of all vertices v such that u can be reached from v.

For two non-adjacent vertices s, t of a directed graph G, a set $S \subseteq V(G) \setminus \{s, t\}$ is said to be an $s - t$ *separator* if $t \notin R_{G-S}^+(s)$. An $s - t$ separator S is *minimal* if no proper subset $S' \subset S$ is an $s - t$ separator.

The notion of important separators was introduced by Marx [13] and generalized for directed graphs in [7]. We need a special variant of this notion. Let G be a directed graph, and let s, t be non-adjacent vertices of G. A minimal $s - t$ separator is an *important $s - t$ separator* if there is no $s - t$ separator S' with $|S'| \leq |S|$ and $R_{G-S}^-(t) \subset R_{G-S'}^-(t)$. The following lemma is a variant of Lemma 4.1 of [7]. Notice that to obtain it, we should replace the directed graph in Lemma 4.1 of [7] by the graph obtained from it by reversing the direction of all arcs.

▶ **Lemma 1** ([7]). *Let G be a directed graph with n vertices, and let s, t be non-adjacent vertices of G. Then for every $h \geq 0$, there are at most 4^h important $s - t$ separators of size at most h. Furthermore, all these separators can be enumerated in time $O(4^h \cdot n^{O(1)})$.*

As further we are interested in the parameterized complexity of Dir-AKC, we show first NP-hardness of the problem (the proof is given in [5]).

▶ **Theorem 2.** *For any $k \geq 1$,* Dir-AKC *is* NP*-complete, even for planar DAGs of maximum degree at most $k + 2$.*

We conclude this section by the simple observation that Dir-AKC is in XP when parameterized by the number of anchors b. For a directed graph G with n vertices, we can consider all the at most n^b possibilities to choose the anchors, and then recursively delete non-anchor vertices that have the in-degree at most $k - 1$. Trivially, if we obtain a directed graph with at least p vertices for some selection of the anchors, we have a solution and otherwise we can answer NO.

3 Dir-AKC parameterized by the size of the core

In this section we consider the Dir-AKC problem for fixed k when p is the parameter, and obtain the following dichotomy: If $k = 1$ then the Dir-AKC problem is FPT parameterized by p, otherwise for $k \geq 2$ it is W[1]-hard parameterized by p.

▶ **Theorem 3.** *For $k = 1$, the* Dir-AKC *problem is solvable in time $2^{O(p)} \cdot n^2 \log n$ on digraphs with n vertices.*

Proof. The proof is constructive, and we describe an FPT algorithm for the problem. Without loss of generality, we assume that $b < p \le n$.

We apply the following preprocessing rule reducing the instance to an acyclic graph. Let $C_1, \dots, C_r$ be the non-trivial strongly connected components of G, i.e., $|V(C_i)| \ge 2$ for $i \in \{1, \dots, r\}$. Note that for each $i \in \{1, \dots, r\}$ and any $v \in V(C_i)$, $d^-_{C_i}(v) \ge 1$. By making use of Tarjan's algorithm [18], $C_1, \dots, C_r$ can be found in linear time. Let $R = R^+_G\Big(\bigcup_{i=1}^r V(C_i)\Big)$ be the set of vertices reachable from these strongly connected components. Then every $v \in R$ satisfies $d^-_{G[R]}(v) \ge 1$. If $|R| \ge p$, then $H = G[R]$ is an anchored 1-core of size at least p for the empty set of anchors. If $b \ge p - |R| > 0$, then we select in $V(G) \setminus R$ any arbitrary $b' = p - |R|$ vertices $a_1, \dots, a_{b'}$. In this case we output the set of anchors $A = \{a_1, \dots, a_{b'}\}$ and the graph $H = G[A \cup R]$. Otherwise, if $b < p - |R|$, we set $G' = G - R$ and $p' = p - |R|$ and consider a new instance of Dir-AKC with the graph G' and the parameter p'.

To see that the rule is safe, it is sufficient to observe that a set of anchors A and a subgraph H' of size at least p' is a solution of the obtained instance if and only if $(A, H = G[V(H') \cup R])$ is a solution for the original problem. Let us remark that the preprocessing rule can be easily performed in time $O(n^2)$.

From now we can assume that G has no non-trivial strongly connected components, i.e., G is a directed acyclic graph. Denote by $S = \{s_1, \dots, s_h\}$ the set of sources of G. If $|S| \le b$, then set $A = S$. In this case, we output the pair $(A, H = G)$. The pair (A, H) is a solution because every vertex $v \in V(G) \setminus S$ satisfies $d^-_G(v) \ge 1$. It remains to consider the case when $|S| > b$. For $i \in \{1, \dots, h\}$, let $R_i = R^+_G(s_i)$. Then $V(G) = R^+_G(S) = \bigcup_{i=1}^h R_i$. Without loss of generality, we can assume that every anchored vertex is from S. Indeed, if s_i is an anchor, then each vertex of R_i can be included in a solution. Hence for every anchor $a \in R_j \setminus \{s_j\}$, we can delete this anchor from A and replace it by s_j. Since we can choose anchors only from S, we are able to reduce the problem to Partial Set Cover.

> Partial Set Cover
> *Input* : A collection $X = \{X_1, \dots, X_r\}$ of subsets of a finite n-element set U and positive integers p, b.
> *Parameter*: p.
> *Question*: Are there at most b subsets $X_{i_1}, \dots, X_{i_b}$, $1 \le i_1 < \dots < i_b \le r$, covering at least p elements of U, i.e., $|\bigcup_{j=1}^b X_{i_j}| \ge p$?

Bläser [3] showed that Partial Set Cover is FPT parameterized by p and can be solved in time $O(2^{O(p)} \cdot rn \log n)$. For Dir-AKC, we consider the collection of subsets $\{R_1, \dots, R_r\}$ of $V(G)$. If we can select at most b subsets $R_{i_1}, \dots, R_{i_b}$ such that $|\cup_{j=1}^b R_{i_j}| \ge p$, we return the solution with anchors $A = \{s_{i_1}, \dots, s_{i_b}\}$ and $H = G[\bigcup_{j=1}^b R_{i_j}]$. Otherwise, we return a NO-answer.

Because our preprocessing can be done in time $O(n^2)$ and Partial Set Cover is solvable in time $2^{O(p)} \cdot n^2 \log n$, we conclude that the total running time is $2^{O(p)} \cdot n^2 \log n$. ◀

Now we complement Theorem 3 by showing that for $k \ge 2$, Dir-AKC becomes hard parameterized by the core size (the proof is in [5]).

▶ **Theorem 4.** *For any fixed $k \ge 2$, the* Dir-AKC *problem is* W[1]*-hard parameterized by p, even for DAGs.*

4 Dir-AKC on graphs of bounded degree

In this section we show that DIR-AKC problem is FPT parameterized by $\Delta + p$ if $k \geq \frac{\Delta}{2}$.

In our algorithms we need to check the existence of solutions for DIR-AKC that have bounded size. It can be observed that if we are interested in solutions (A, H) such that $p \leq |V(H)| \leq q$, then for every positive q, we can express this problem in First Order Logic. It was proved by Seese [16] that any graph problem expressible in First Order Logic can be solved in linear time on (directed) graphs of bounded degree. Later this result was extended for much more rich graph classes (see [11]). These meta theorems are very general, but do not provide good upper bounds on the running time for particular problems. Hence, we give the following lemma. Our algorithms use the random separation technique due to Cai et al. [4] (which is a variant of the color coding method introduced by Alon et al. [1]) .

▶ **Lemma 5.** *There is a randomized algorithm with running time $2^{O(\Delta q)} \cdot n$ that for an instance of* DIR-AKC *with an n-vertex directed graph of maximum degree at most Δ and a positive integer $q \geq p$, either returns a solution (A, H) with $V(H) \geq p$ or gives the answer that there is no solution with $|V(H)| \leq q$. Furthermore, the algorithm can be derandomized, and the deterministic variant runs in time $2^{O(\Delta q)} \cdot n \log n$.*

Proof. Consider an instance of DIR-AKC with an n-vertex directed graph G of maximum degree at most Δ. We assume that $b \leq p \leq n$. For given $q \geq p$, to decide if G contains a solution of size at most q, we do the following.

We color each vertex of G uniformly at random with probability $\frac{1}{2}$ by one of two colors, say red or blue. Let R be the set of vertices colored red. Observe that if there is a solution (A, H) with $|V(H)| \leq q$, then with probability at least $\frac{1}{2^q}$ all vertices of H are colored red and with probability at least $\frac{1}{2^{\Delta q}}$ all in- and out-neighbors of the vertices of H that are outside of H are colored blue. Using this observation, we assume that H is the union of some weakly connected components of the graph $G[R]$ induced by red vertices.

In time $O(\Delta n)$ we find all weakly connected components of $G[R]$. If there is a component C with at least $b+1$ vertices of in-degree at most $k-1$ (in C), then we discard this component as it cannot be a part of any solution. Denote by $C_1, \ldots, C_r$ the remaining components. For $i \in \{1, \ldots, r\}$, let $A_i = \{v \in V(C_i) | d^-_{C_i}(v) < k\}$, $b_i = |A_i|$ and $p_i = |V(C_i)|$.

Thus everything boils down to the problem of finding a set $I \subseteq \{1, \ldots, r\}$ such that $\sum_{i \in I} b_i \leq b$ and $\sum_{i \in I} p_i \geq p$. But this is the well known KNAPSACK problem, which is solvable in time $O(bn)$ by dynamic programming. If we obtain a solution I, then we output (A, H), where $A = \cup_{i \in I} A_i$ and $H = G[\cup_{i \in I} V(C_i)]$. Otherwise, we return a NO-answer. Notice that this algorithm can also find a solution (A, H) with $|V(H)| > q \geq p$.

It remains to observe that for any positive number $\alpha < 1$, there is a constant c_α such that after running our randomized algorithm $c_\alpha \cdot 2^{\Delta q}$ times, we either find a solution (A, H) or can claim that with probability α that it does not exist.

This algorithm can be derandomized by the technique proposed by Alon et al. [1]: replace the random colorings by a family of at most $2^{O(\Delta q)} \cdot \log n$ hash functions which are known to be constructible in time $2^{O(\Delta q)} \cdot n \log n$. ◀

Our next aim is to prove that for $k > \Delta/2$ the DIR-AKC problem is FPT when parameterized by $\Delta + b$.

▶ **Lemma 6.** *Let Δ be a positive integer. If $k > \Delta/2$, then the* DIR-AKC *problem can be solved in time $2^{O(\Delta^2 b)} \cdot n \log n$ for n-vertex directed graphs of maximum degree at most Δ.*

Proof. Suppose (A, H) is a solution for the Dir-AKC problem. Let us observe that because $k > \Delta/2$, for every vertex $v \in V(H)\setminus A$, we have $d^-_H(v) > d^+_H(v)$. Recall that for any directed graph, the sum of in-degrees equals the sum of out-degrees. Then

$$\sum_{v\in V(H)\setminus A} (d^-_H(v) - d^+_H(v)) = \sum_{v\in A}(d^+_H(v) - d^-_H(v)).$$

Since for every vertex $v \in V(H) \setminus A$, $d^-_H(v) - d^+_H(v) \geq 1$, we have that

$$|V(H) \setminus A| \leq \sum_{v\in V(H)\setminus A} (d^-_H(v) - d^+_H(v)).$$

On the other hand, $d^+_H(v) - d^-_H(v) \leq \Delta$, and we arrive at

$$|V(H) \setminus A| \leq \sum_{v\in V(H)\setminus A} (d^-_H(v) - d^+_H(v)) = \sum_{v\in A}(d^+_H(v) - d^-_H(v)) \leq \Delta|A|.$$

Hence, $|V(H)| \leq (\Delta+1)|A| \leq (\Delta+1)b$. Using this observation, we can solve the Dir-AKC problem as follows. If $p > (\Delta+1)b$, then we return a NO-answer. If $p \leq (\Delta+1)b$, we apply Lemma 5 for $q = (\Delta+1)b$, and solve that problem in time $2^{O(\Delta^2 b)} \cdot n \log n$. ◀

Now we show that if $k = \frac{\Delta}{2}$ then the Dir-AKC problem is FPT parameterized by $\Delta+p$.

▶ **Lemma 7.** *Let Δ be a positive integer. If $k = \Delta/2$, then the* Dir-AKC *problem can be solved in time $2^{O(\Delta^3 b+\Delta^2 bp)} \cdot n^{O(1)}$ for n-vertex directed graphs of maximum degree at most Δ.*

Proof. We describe an FPT algorithm. Consider an instance of the Dir-AKC problem. Without loss of generality we assume that $b < p \leq n$.

We apply the following preprocessing rule. Suppose that G has a (weakly) connected component C such that for any $v \in V(C)$, $d^-_C(v) = d^+_C(v) = k$. If $b \geq p - |V(C)|$, then we choose a set A of $b' = p - |V(C)|$ vertices arbitrary in $V(G) \setminus V(C)$. Then we return a YES-answer, as the anchors A and $H = G[A \cup V(C)]$ is a solution. Otherwise, if $b < p - |V(C)|$, we let $G' = G - V(C)$ and $p' = p - |V(C)|$. Now we consider a new instance of the problem with the graph G' and the parameter p'. To see that the rule is safe, it is sufficient to observe that a set of anchors A and a subgraph H' of size at least p' is a solution of the obtained instance if and only if A and $H = G[V(H') \cup V(C)]$ is a solution for the original problem. From now we assume that G has no such components.

We need the following claim.

Claim A. *If an instance of the* Dir-AKC *problem has a core with at least $(\Delta p + 1)b + 1$ vertices, then it has a solution (A, H) with the following property: there is a vertex $t \in V(H) \setminus A$ reachable in H from any vertex of H. Moreover, for each vertex v of H, there is a path from v to t with all vertices except v in $V(H) \setminus A$.*

Proof of Claim A. Let (A, H') be a solution with the set of anchors A and such that $V(H') > (\Delta p + 1)b$.

We show that $V(H') = R^+_{H'}(A)$, i.e., all vertices of H' are reachable from the anchors. To obtain a contradiction, suppose that there is a vertex $u \in V(H')$ such that $u \notin R^+_{H'}(A)$. Let $U = R^-_{H'}(u)$, i.e., U is the set of vertices from which we can reach u. Clearly, $A \cap U = \emptyset$. Therefore, $d^-_{H'}(v) \geq k = \Delta/2$ for $v \in U$. Notice that for a vertex $v \in U$, $N^-_{H'}(v) \subseteq U$ by the definition. Hence, $d^-_{G[U]}(v) \geq k = \Delta/2$ for $v \in U$. Because the sum of in-degrees equals the sum of out-degrees, for every vertex $v \in U$, we have that $d^-_{G[U]}(v) = d^+_{G[U]}(v) = k = \Delta/2$. Then $C = G[U]$ is a component of G such that for every $v \in V(C)$, $d^-_C(v) = d^+_C(v) = k$, but such components are excluded by the preprocessing; a contradiction.

Observe now that if $d^-_{H'}(v) < d^+_{H'}(v)$, then $d^-_{H'}(v) < k$ and thus $v \in A$. Hence, by adding at most Δb (maybe multiple) arcs from $V(H') \setminus A$ to A, joining the vertices $v \in V(H')$ of degrees $d^-_{H'}(v) > d^+_{H'}(v)$ with vertices of degrees $d^-_{H'}(v) < d^+_{H'}(v)$, we can transform H' into a disjoint union of directed Eulerian graphs. Since $V(H') = R^+_{H'}(A)$, each of these directed Eulerian graphs contains at least one vertex of A. Thus the set of arcs of H' can be covered by at most Δb arc-disjoint directed walks, each walk starting from a vertex of A and never coming back to A. Because $d^-_{H'}(v) \geq k$ for $v \in V(H') \setminus A$, we have that $|E(G')| \geq k(|V(H')| - b) > \Delta kbp$. Then there is a walk W with at least $kp + 1$ arcs. Let $a \in A$ be the first vertex of W and let t be the last vertex of the walk. The walk W visits a only once, t and all other vertices of W are visited at most k times. We conclude that W has at least p vertices.

Let $R = R^-_{H'-A}(t)$ and let $A' = \{a \in A \mid N^+_{H'}(a) \cap R \neq \emptyset\}$. Consider $H = G[R \cup A']$. Since $V(W) \subseteq V(H)$, $|V(H)| \geq p$. For any $v \in V(H) \setminus A$, the in-neighbors of v in H' are in H by the construction and, therefore, $d^-_H(v) \geq k$. It remains to observe that to select at most b anchors, we take $A' \subseteq V(H)$. ◀

Using Claim A, we proceed with our algorithm. We try to find a solution such that H has at most $q = (\Delta p + 1)b$ vertices by applying Lemma 5. It takes time $O(2^{O(\Delta^2 bp)} \cdot n \log n)$. If we obtain a solution, then we return it and stop. Otherwise, we conclude that every core contains at least $(\Delta p + 1)b + 1$ vertices. By Claim A, we can search for a solution H with a non-anchor vertex t which is reachable from all other vertices of H by directed paths avoiding A. Notice that since t is a non-anchor vertex, we have that $d^-_G(t) \geq k$. We try at most n possibilities for all possible choices of t, and solve our problem for each choice. Clearly, if we get a YES-answer for one of the choices, we return it and stop. Otherwise, if we fail, we return a NO-answer.

From now we assume that we already selected t. We denote by G' the graph obtained from G by adding an artificial source vertex s joined by arcs with all the vertices $v \in V(G)$ with $d^-_G(v) < k$. Observe that $(s, t) \notin E(G')$.

Suppose that (A, H) is a solution with the set of anchors A such that $t \in V(H) \setminus A$ is reachable in H from any vertex of H by a path with all inner vertices in $V(H) \setminus A$. Denote by $\delta_{G'}(H)$ the set $\{v \in V(H) \mid N^-_{G'}(v) \setminus V(H) \neq \emptyset\}$, i.e., $\delta_{G'}(H)$ contains vertices that have in-neighbors outside H. We need a chain of claims about the structure of H in G'.

Claim B. $|\delta_{G'}(H) \setminus A| \leq \Delta b$.

Proof of Claim B. Let $X = \{v \in V(H) \mid d^-_H(v) \geq k \text{ and } d^+_H(v) < k\}$, $Y = \{v \in V(H) \mid d^-_H(v) = d^+_H(v) = k\}$ and $Z = \{v \in V(H) \mid d^-_H(v) < k\}$. Clearly,

$$\sum_{v \in X}(d^-_H(v) - d^+_H(v)) + \sum_{v \in Y}(d^-_H(v) - d^+_H(v)) = \sum_{v \in Z}(d^+_H(v) - d^-_H(v))$$

Observe that $d^-_H(v) - d^+_H(v) \geq 1$ for $v \in X$, $d^-_H(v) - d^+_H(v) = 0$ for $v \in Y$ and $d^+_H(v) - d^-_H(v) \leq \Delta$ for $v \in Z$. Hence, $|X| \leq \Delta|Z|$. If $d^-_H(v) < k$ for $v \in V(H)$, then $v \in A$. It follows that $Z \subseteq A$ and $|Z| \leq b$. We have $|X| \leq \Delta b$. Consider a vertex $v \in \delta_{G'}(H) \setminus A$. It has at least one in-neighbor outside H in G and $d^-_H(v) \geq k$. Then $d^+_H(v) < k$ and $v \in X$. We conclude that $\delta_{G'}(H) \setminus A \subseteq X$ and $|\delta_{G'}(H) \setminus A| \leq \Delta b$. ◀

Claim C. *There is an $s-t$ separator S in G' of size at most $(\Delta(k-1)+1)b$ such that $V(H)\setminus A \subseteq R^-_{G'-S}(t)$.*

Proof of Claim C. Let $S = \Big(\delta_{G'}(H)\cap A\Big) \cup \Big(\bigcup_{v\in\delta_{G'}(H)\setminus A}(N^-_G(v)\setminus V(H)\Big)$, i.e., the set containing all anchors that are in $\delta_{G'}$, and for each non-anchor vertex of $\delta_{G'}$ containing all its in-neighbors outside of H. Consider a directed (s,t)-path P in G'. Let v be the first vertex in P that is in $V(H)$ and let u be its predecessor in P. If $v\in A$, then $v\in S$. If $v\notin A$, then $u\neq s$ as H has no non-anchor vertices with in-degree at most $k-1$ in G. Then $u\in S$. We conclude that each (s,t)-path contains a vertex of S, i.e., this set is an $s-t$ separator.

Observe that $V(H)\setminus A\subseteq R^-_{G'-S}(t)$ by the definition of S and the fact that t can be reached from any vertex of H in this graph by a path with all inner vertices in $V(H)\setminus A$.

It remains to show that $|S|\leq(\Delta(k-1)+1)b$. By Claim B, $|\delta_{G'}(H)\setminus A|\leq\Delta b$. A vertex $v\in\delta_{G'}(H)\setminus A$ has at least one out-neighbor in H because t is reachable from v. Then v has at most $k-1$ in-neighbors outside H. Hence $|S|\leq|A|+(k-1)(\delta_{G'}(H)\setminus A)\leq(\Delta(k-1)+1)b$. ◀

Now we can prove the following claim about important $s-t$ separators in G'.

Claim D. *There is an important $s-t$ separator S^* of size at most $(\Delta(k-1)+1)b$ in G' such that $V(H)\subseteq R^-_{G'-S^*}(t)\cup S^*$.*

Proof of Claim D. By Claim C, there is an $s-t$ separator S' in G' of size at most $(\Delta(k-1)+1)b$ such that $V(H)\setminus A\subseteq R^-_{G'-S'}(t)$. Notice that S' not necessary a minimal separator, but there is a minimal $s-t$ separator $S\subseteq S'$. Clearly, $|S|\leq(\Delta(k-1)+1)b$.

We show that $V(H)\subseteq R^-_{G'-S}(t)\cup S$. Because $R^-_{G'-S'}(t)\subseteq R^-_{G'-S}(t)$, we have that $V(H)\setminus A\subseteq R^-_{G'-S}(t)$. Also if an anchor a is in $R^-_{G'-S'}(t)$, then $a\in R^-_{G'-S}(t)$. Let $a\in A\cap S'$. If $a\in A\cap S$, then $a\in R^-_{G'-S}(t)\cup S$. If $a\notin S$, then by Claim C, a has an out-neighbor $v\in R^-_{G'-S'}(t)$ and in this case we have $a\in R^-_{G'-S}(t)$.

It remains to observe that there is an important $s-t$ separator S^* such that $|S^*|\leq|S|\leq(\Delta(k-1)+1)b$ and $R^-_{G'-S}(t)\subseteq R^-_{G'-S^*}(t)$. Therefore, $V(H)\subseteq R^-_{G'-S}(t)\cup S\subseteq R^-_{G'-S^*}(t)\cup S^*$. ◀

The next step of our algorithm is to check all important $s-t$ separators in G' of size at most $(\Delta(k-1)+1)$. By Lemma 1, there are at most $4^{(\Delta(k-1)+1)b}$ important $s-t$ separators and they can be listed in time $2^{O(\Delta^2 b)}\cdot n^c$. For each important $s-t$ separator S^*, we consider the set of vertices $U=R^-_{G'-S^*}(t)\cup S^*$ and decide whether there is a solution such that $V(H)\subseteq U$. If we have a solution for some S^*, then we return a YES-answer and stop. Otherwise, if we fail to find such a solution for all important separators, we use Claim D to deduce that there is no solution.

From now on, we assume that an important $s-t$ separator S^* is given and that $U=R^-_{G'-S^*}(t)\cup S^*$. In what follows, we describe a procedure of finding a solution with $V(H)\subseteq U$.

Denote by D the set $\{v\in U\mid d^-_G(v)>0\}$. We need the following observation.

Claim E. *Set D contains at most $(\Delta+1)(\Delta(k-1)+1)b$ vertices.*

Proof of Claim E. Let $Q=G[U]$. Let $X=\{v\in V(Q)\mid d^-_Q(v)\geq k \text{ and } d^+_Q(v)<k\}$, $Y=\{v\in V(Q)\mid d^-_Q(v)=d^+_Q(v)=k\}$ and $Z=\{v\in V(Q)\mid d^-_Q(v)<k\}$. Clearly,

$$\sum_{v\in X}(d^-_Q(v)-d^+_Q(v))+\sum_{v\in Y}(d^-_Q(v)-d^+_Q(v))=\sum_{v\in Z}(d^+_Q(v)-d^-_Q(v))$$

Observe that $d^-_Q(v) - d^+_Q(v) \geq 1$ for $v \in X$, $d^-_Q(v) - d^+_Q(v) = 0$ for $v \in Y$ and $d^+_Q(v) - d^-_Q(v) \leq \Delta$ for $v \in Z$. Hence, $|X| \leq \Delta|Z|$.

Recall that G' is obtained from G by joining s with all vertices of in-degree at most $k-1$. Since S^* is an $s-t$ separator, if for $v \in U$, $d^-_Q(v) < k$, then $v \in S^*$. Hence, $Z \subseteq S^*$ and $|Z| \leq |S^*| \leq (\Delta(k-1)+1)b$. If for for $v \in U$, $d^-_G(v) > k$, then $v \in X \cup Z$. We conclude that $|D| \leq |X| + |Z| \leq (\Delta+1)|Z| \leq (\Delta+1)(\Delta(k-1)+1)b$. ◀

Recall that set $\delta_{G'}(H)$ contains vertices of H that have in-neighbors outside of H. If $v \in \delta_{G'}(H) \setminus A$, then it has at least k in-neighbors in H and at least one in-neighbor outside H. Notice that $s \notin N^-_{G'}(v)$ because $d^-_G(v) \geq d^-_H(v) \geq k$. Hence, $d^-_G(v) > k$. Because $V(H) \subseteq U$, $\delta_{G'}(H) \setminus A \subseteq D$. By Claim C, $|\delta_{G'}(H) \setminus A| \leq \Delta b$, and by Claim E, $|D| \leq (\Delta+1)(\Delta(k-1)+1)b$. We consider all at most $2^{(\Delta+1)(\Delta(k-1)+1)b}$ possibilities to select $\delta_{G'}(H) \setminus A$. For each choice of $\delta_{G'}(H) \setminus A$, we guess the arcs that join the vertices that are outside H with the vertices of $\delta_{G'}(H) \setminus A$ and delete them. Denote the graph obtained from G by F. Recall that from each vertex v of $\delta_{G'}(H) \setminus A$, there is a directed path to t that avoids A. Hence, v has at least one out-neighbor in H and at most $\Delta - 1$ in-neighbors in G. Also v has at least k in-neighbors in H, and we delete at most $d^-_G(v) - k$ arcs. Therefore, for v we choose at most $k-1$ arcs out of at most $\Delta - 1$ arcs. We can upper bound the number of possibilities for v by $2^{\Delta-1}$, and the total number of possibilities for $\delta_{G'}(H) \setminus A$ by $2^{(\Delta-1)\Delta b}$.

Observe that (A, H) is a solution for the new instance of Dir-AKC, where G is replaced by F for a correct guess of the deleted arcs. Also each solution for the new instance provides a solution for the graph G, because if we put deleted arcs back, then we can only increase the in-degrees. Hence, we can check for each possible choice of the set of deleted arcs, whether the new instance has a solution. If for some choice we obtain a solution, then we return a YES-answer. Otherwise, if we fail for all choices, then we return a NO-answer. Further we assume that F is given.

Denote by F' the graph obtained from F by the addition of a vertex s joined by arcs with all the vertices $N^+_{G'}(s)$. Now $\delta_{F'}(H) = \{v \in V(H) \mid N^-_{F'}(v) \setminus V(H) \neq \emptyset\}$. By the choice of F, $\delta_{F'}(H) = \delta_{G'}(H) \cap A$ and, therefore, $|\delta_{F'}(H)| \leq b$. Also $\delta_{F'}(H)$ is an $s-t$ separator in F' by Claim C.

Now we can prove the following.

Claim F. *There is an important $s-t$ separator $\hat{S}$ of size at most b in F' such that $(\hat{S}, G[R^-_{F'-\hat{S}}(t) \cup \hat{S}])$ is a solution for the instance of the* Dir-AKC *problem for the graph G.*

Proof of Claim F. Let $U = R^-_{F'-\hat{S}}(t) \cup \hat{S}$. It was already observed that $\delta^*_{G'}(H)$ is an $s-t$ separator in F' of size at most b. Then there is a minimal $s-t$ separator $S \subseteq \delta^*_{G'}(H)$. Clearly, $|S| \leq b$.

As before in the proof of Claim D, we show that $V(H) \subseteq R^-_{F'-S}(t) \cup S$. Because for any vertex v of H, there is a directed (v,t) path with all inner vertices in $V(H) \setminus A$, $V(H) \setminus A \subseteq R^-_{F'-\delta_{F'}(H)}(t)$. Because $R^-_{F'-\delta_{F'}(H)}(t) \subseteq R^-_{F'-S}(t)$ we have $V(H) \setminus A \subseteq R^-_{F'-S}(t)$. Also if $a \in A$ is in $R^-_{F'-\delta_{F'}(H)}(t)$, then $a \in R^-_{F'-S}(t)$. Let $a \in A \cap \delta_{F'}(H)$. Trivially, if $a \in A \cap S$, then $a \in R^-_{F'-S}(t) \cup S$. If $a \notin S$, then a has an out-neighbor $v \in R^-_{F'-\delta_{F'}(H)}(t)$ and $a \in R^-_{F'-S}(t)$. Then there is an important $s-t$ separator $\hat{S}$ such that $|\hat{S}| \leq |S| \leq b$ and $R^-_{F'-S}(t) \subseteq R^-_{F'-\hat{S}}(t)$. Therefore, $V(H) \subseteq R^-_{F'-S}(t) \cup S \subseteq R^-_{F'-S^*}(t) \cup S^*$, and $|U| \geq p$.

It remains to observe that s is adjacent to all vertices of G with in-degrees at most $k-1$ and S^* is an $s-t$ separator. It immediately follows that for any vertex $v \in R^-_{F'-S^*}(t)$, $d^-_{F(U)}(v) \geq k$. Then $(\hat{S}, G[R^-_{F'-\hat{S}}(t) \cup \hat{S}])$ is a solution. ◀

The final step of our algorithm is to enumerate all important $s-t$ separators $\hat{S}$ of size at most b in F', which number by Lemma 1 is at most 4^b, and for each $\hat{S}$, check whether $(\hat{S}, G[R^-_{F'-\hat{S}}(t) \cup \hat{S}])$ is a solution. Recall that all these separators can be listed in time $2^{O(b)} \cdot n^c$. We return a YES-answer if we obtain a solution for some important separator, and a NO-answer otherwise.

To complete the proof, let us observe that each step of the algorithm runs either in polynomial or FPT time. Particularly, the preprocessing is done in time $O(\Delta n)$. Then we check the existence of a solution of a bounded size in time $2^{O(\Delta^2 bp)} \cdot n \log n$. Further we consider at most n possibilities to choose t. For each t, we consider at most $4^{(\Delta(k-1)+1)b}$ important $s-t$ separators S^*. Recall, that they can be listed in time $2^{O(\Delta^2 b)} \cdot n^c$ for some constant c. Then for each S^*, we have at most $2^{(\Delta+1)(\Delta(k-1)+1)b+(\Delta-1)}$ possibilities to construct F, and it can be done in time $2^{O(\Delta^3 b)} + O(\Delta n)$. Finally, there are at most 4^b important $s-t$ separators $\hat{S}$ and they can be listed in time $2^{O(b)} \cdot n$ for some c. We conclude that the total running time is $2^{O(\Delta^3 b + \Delta^2 bp)} \cdot n^c$ for some constant c. ◀

Combining Lemmas 6 and 7, we obtain the following theorem.

▶ **Theorem 8.** *Let Δ be a positive integer. If $k \geq \frac{\Delta}{2}$, then the* Dir-AKC *problem can be solved in time $2^{O(\Delta^3 b + \Delta^2 bp)} \cdot n^{O(1)}$ for n-vertex directed graphs of maximum degree at most Δ.*

Theorems 3 and 8 give the next corollary.

▶ **Corollary 9.** *The* Dir-AKC *problem can be solved in time $2^{O(bp)} \cdot n^{O(1)}$ for n-vertex directed graphs of maximum degree at most 4.*

5 Conclusions

We proved that Dir-AKC is NP-complete even for planar DAGs of maximum degree at most $k+2$. It was also shown that Dir-AKC is FPT when parameterized by $p+\Delta$ for directed graphs of maximum degree at most Δ whenever $k \geq \Delta/2$. It is natural to ask whether the problem is FPT for other values k. This question is interesting even for the special case $\Delta = 5$ and $k = 2$.

For the special case of directed acyclic graphs (DAGs) we understand the complexity of the problem much better. Theorem 4 showed that Dir-AKC on DAGs is W[1]-hard parameterized by p for every fixed $k \geq 2$, when the degree of the graph is not bounded. We now show the following theorem (the proof is in [5]) that gives W[2]-hardness of Dir-AKC when parameterized by the number of anchors b (recall that we can always assume that $b \leq p$).

▶ **Theorem 10.** *For any $\Delta \geq 3$ and any positive $k < \frac{\Delta}{2}$,* Dir-AKC *is* W[2]*-hard (even on DAGs) when parameterized by the number of anchors b on graphs of maximum degree at most Δ.*

The complexity of Dir-AKC parameterized by b on DAGs for the case of $k \geq \frac{\Delta}{2}$ is left open. However we can show that Dir-AKC is FPT on DAGs of maximum degree Δ, when parameterized by $\Delta + p$ (the proof is in [5].)

▶ **Theorem 11.** *For any positive integers p and Δ,* Dir-AKC *can be solved in time $2^{O(\Delta p)} \cdot n^2 \log n$ for n-vertex DAGs of maximum degree at most Δ.*

Let us remark that this result can be easily extended for any class of directed acyclic graphs $\mathcal{G}$ such that the corresponding class of underlaying graphs $\{G^*|G \in \mathcal{G}\}$ has (locally) bounded expansion by making use of the results by Dvorak et al. [11]. Finally, what happens when the input graph is planar? We know that the problem is NP-complete on planar graphs for fixed $k \geq 1$ and maximum degree $k+2$. Is the problem FPT on planar directed graphs when parameterized by the size of the core p?

References

1 N. Alon, R. Yuster, and U. Zwick. Color-coding. *J. ACM*, 42(4):844–856, 1995.
2 K. Bhawalkar, J. M. Kleinberg, K. Lewi, T. Roughgarden, and A. Sharma. Preventing unraveling in social networks: The anchored k-core problem. In *ICALP '12*, vol. 7392 of *Lecture Notes in Computer Science*, pages 440–451, 2012.
3 M. Bläser. Computing small partial coverings. *IPL.*, 85(6):327–331, 2003.
4 L. Cai, S. M. Chan, and S. O. Chan. Random separation: A new method for solving fixed-cardinality optimization problems. In *IWPEC'06*, vol. 4169 of *Lecture Notes in Computer Science*, pages 239–250, 2006.
5 Ra. H. Chitnis, F. V. Fomin, and P. A. Golovach. Parameterized complexity of the anchored k-core problem for directed graphs. *CoRR*, abs/1304.5870, 2013.
6 R. H. Chitnis, F. V. Fomin, and P. A. Golovach. Preventing unraveling in social networks gets harder. In *AAAI '13*. AAAI Press, 2013.
7 R. H. Chitnis, M. T. Hajiaghayi, and D. Marx. Fixed-parameter tractability of directed multiway cut parameterized by the size of the cutset. In *SODA '12*, pages 1713–1725. SIAM, 2012.
8 M. Chwe. Structure and Strategy in Collective Action 1. *American Journal of Sociology*, 105(1):128–156, 1999.
9 M. Chwe. Communication and Coordination in Social Networks. *The Review of Economic Studies*, 67(1):1–16, 2000.
10 R. G. Downey and M. R. Fellows. *Parameterized Complexity.* Springer-Verlag, 1999.
11 Z. Dvorak, D. Král, and R. Thomas. Deciding first-order properties for sparse graphs. In *FOCS*, pages 133–142. IEEE Computer Society, 2010.
12 J. Flum and M. Grohe. *Parameterized Complexity Theory.* Texts in Theoretical Computer Science. An EATCS Series. Springer-Verlag, Berlin, 2006.
13 D. Marx. Parameterized graph separation problems. *Theor. Comput. Sci.*, 351(3):394–406, 2006.
14 R. Niedermeier. *Invitation to fixed-parameter algorithms*, volume 31 of *Oxford Lecture Series in Mathematics and its Applications.* Oxford University Press, Oxford, 2006.
15 T.C. Schelling. *Micromotives and Macrobehavior.* WW Norton, 2006.
16 D. Seese. Linear time computable problems and first-order descriptions. *Mathematical Structures in Computer Science*, 6(6):505–526, 1996.
17 S. B. Seidman. Network Structure and Minimum Degree. *Social networks*, 5(3):269–287, 1983.
18 R. E. Tarjan. Depth-first search and linear graph algorithms. *SIAM J. Comput.*, 1(2):146–160, 1972.