Directed Subset Feedback Vertex Set is Fixed-Parameter Tractable[*]

Rajesh Chitnis[†1], Marek Cygan[‡2], MohammadTaghi Hajiaghayi[§1] and Dániel Marx[¶3]

[1]Department of Computer Science, University of Maryland at College Park, USA. Email: {rchitnis,hajiagha}@cs.umd.edu
[2]Institute of Informatics, University of Warsaw, Poland. Email: cygan@mimuw.edu.pl
[3]Institute for Computer Science and Control, Hungarian Academy of Sciences (MTA SZTAKI), Budapest, Hungary. Email: dmarx@cs.bme.hu

November 30, 2014

Abstract

Given a graph G and an integer k, the FEEDBACK VERTEX SET (FVS) problem asks if there is a vertex set T of size at most k that hits all cycles in the graph. The first fixed-parameter algorithm for FVS in undirected graphs appeared in a monograph of Mehlhorn in 1984. The fixed-parameter tractability status of FVS in directed graphs was a long-standing open problem until Chen et al. (STOC '08, JACM '08) showed that it is fixed-parameter tractable by giving a $4^k k! n^{O(1)}$ time algorithm. There are two subset versions of this problems: we are given an additional subset S of vertices (resp., edges) and we want to hit all cycles passing through a vertex of S (resp. an edge of S); the two variants are known to be equivalent in the parameterized sense. Recently, the SUBSET FEEDBACK VERTEX SET problem in undirected graphs was shown to be FPT by Cygan et al. (ICALP '11, SIDMA '13) and independently by Kakimura et al. (SODA '12). We generalize the result of Chen et al. (STOC '08, JACM '08) by showing that SUBSET FEEDBACK VERTEX SET in directed graphs can be solved in time $2^{O(k^3)} n^{O(1)}$, i.e., FPT parameterized by size k of the solution. By our result, we complete the picture for feedback vertex set problems and their subset versions in undirected and directed graphs.

The technique of random sampling of important separators was used by Marx and Razgon (STOC '11, SICOMP '14) to show that UNDIRECTED MULTICUT is FPT and was generalized by Chitnis et al. (SODA '12, SICOMP '13) to directed graphs to show that DIRECTED MULTIWAY CUT is FPT. Besides proving the fixed-parameter tractability of DIRECTED SUBSET FEEDBACK VERTEX SET, we reformulate the random sampling of important separators technique in an abstract way that can be used for a general family of transversal problems. We believe this general approach will be useful for showing the fixed-parameter tractability of other problems

in directed graphs. Moreover, we modify the probability distribution used in the technique to achieve better running time; in particular, this gives an improvement from $2^{2^{O(k)}}$ to $2^{O(k^2)}$ in the parameter dependence of the DIRECTED MULTIWAY CUT algorithm of Chitnis et al. (SODA '12, SICOMP '13).

1   Introduction

The FEEDBACK VERTEX SET (FVS) problem has been one of the most extensively studied problems in the parameterized complexity community. Given a graph G and an integer k, it asks if there is a set $T \subseteq V(G)$ of size at most k which hits all cycles in G. The FVS problem in both undirected and directed graphs was shown to be NP-hard by Karp [27]. A generalization of the FVS problem is SUBSET FEEDBACK VERTEX SET (SFVS): given a subset $S \subseteq V(G)$ (resp., $S \subseteq E(G)$), find a set $T \subseteq V(G)$ of size at most k such that T hits all cycles passing through a vertex of S (resp., an edge of S). It is easy to see that $S = V(G)$ (resp., $S = E(G)$) gives the FVS problem.

As compared to undirected graphs, FVS behaves quite differently on directed graphs. In particular the trick of replacing each edge of an undirected graph G by arcs in both directions does not work: every feedback vertex set of the resulting digraph is a vertex cover of G and vice versa. Any other simple transformation does not seem possible either and thus the directed and undirected versions are very different problems. This is reflected in the best known approximation ratio for the directed versions as compared to the undirected problems: FVS in undirected graphs has an 2-approximation [1] while FVS in directed graphs has an $O(\log |V(G)| \log \log |V(G)|)$-approximation [19, 44]. The more general SFVS problem in undirected graphs has an 8-approximation [20] while the best-known approximation ratio in directed graphs is $O(\min\{\log |V(G)| \log \log |V(G)|, \log^2 |S|\})$ [19].

Rather than finding approximate solutions in polynomial time, one can look for exact solutions in time that is superpolynomial, but still better than the running time obtained by brute force solutions. In both the directed and the undirected versions of the feedback vertex set problems, brute force can be used to check in time $n^{O(k)}$ if a solution of size at most k exists: one can go through all sets of size at most k. Thus the problem can be solved in polynomial time if the optimum is assumed to be small. In the undirected case, we can do significantly better: since the first FPT algorithm for FVS in undirected graphs by Mehlhorn [36] almost 30 years ago, there have been a number of papers [2, 3, 5, 6, 16, 17, 23, 26, 39, 40] giving faster algorithms and the current fastest (randomized) algorithm runs in time $O^*(3^k)$ [13] (the O^* notation hides all factors that are polynomial in the size of input). That is, undirected FVS is fixed-parameter tractable parameterized by the size of the solution. Recall that a problem is *fixed-parameter tractable* (FPT) with a particular parameter k if it can be solved in time $f(k)n^{O(1)}$, where f is an arbitrary function depending only on k; see [18, 21, 38] for more background. For digraphs, the fixed-parameter tractability status of FVS was a long-standing open problem (almost 16 years) until Chen et al. [8] resolved it by giving an $O^*(4^k k!)$ algorithm. This was recently generalized by Bonsma and Lokshtanov [4] who gave a $O^*(47.5^k k!)$ algorithm for FVS in mixed graphs, i.e., graphs having both directed and undirected edges.

In the more general SUBSET FEEDBACK VERTEX SET problem, an additional subset S of vertices is given and we want to find a set $T \subseteq V(G)$ of size at most k that hits all cycles passing through a vertex of S. In the edge version, we are given a subset $S \subseteq E(G)$ and we want to hit all cycles passing through an edge of S. The vertex and edge versions are indeed known to be equivalent in the parameterized sense in both undirected and directed graphs. Recently, Cygan et al. [15] and independently Kakimura et al. [25] have shown that SUBSET FEEDBACK VERTEX SET in undirected graphs is FPT parameterized by the size of the solution. Our main result is that SUBSET

FEEDBACK VERTEX SET in directed graphs is also fixed-parameter tractable parameterized by the size of the solution:

Theorem 1.1. (main result) SUBSET FEEDBACK VERTEX SET *(*SUBSET-DFVS*) in directed graphs can be solved in time* $O^*(2^{O(k^3)})$.

Our techniques. As a first step, we use the standard technique of *iterative compression* [43] to argue that it is sufficient to solve the compression version of SUBSET-DFVS, where we assume that a solution T of size $k + 1$ is given in the input and we have to find a solution of size k. Our algorithm for the compression problem uses the technique of "random sampling of important separators," which was introduced by Marx and Razgon [35] for undirected MULTICUT and generalized to directed graphs by Chitnis et al. [11] to handle DIRECTED MULTIWAY CUT. We contribute two improvements to this technique on directed graphs. First, we abstract out a framework that allows the clean and immediate application of this technique for various problems. Second, we modify the random selection process to improve the probability of success. In particular, plugging in this improved result to the DIRECTED MULTIWAY CUT algorithm of Chitnis et al. [11], the running time decreases from $O^*(2^{2^k})$ to $O^*(2^{O(k^2)})$ thus giving an exponential improvement.

Theorem 1.2. DIRECTED MULTIWAY CUT *can be solved in time* $O^*(2^{O(k^2)})$, *where k is the number of vertices to be deleted.*

Our generic framework can be used for the following general family of problems. Let $\mathcal{F} = \{F_1, F_2, \ldots, F_q\}$ be a set of subgraphs of a graph G. An $\mathcal{F}$-*transversal* is a set of vertices that intersects every F_i. We consider problems that can be formulated as finding an $\mathcal{F}$-traversal. In particular, we will investigate $\mathcal{F}$-transversal problems satisfying the following property: we say that $\mathcal{F}$ is T-*connected* if for every $i \in [q]$, each vertex of F_i can reach some vertex of T by a walk completely contained in F_i and is reachable from some vertex of T by a walk completely contained in F_i.

$\mathcal{F}$-transversal for T-connected $\mathcal{F}$
Input: A directed graph G, a positive integer k, and a set $T \subseteq V(G)$.
Parameter: k
Question: Does there exist an $\mathcal{F}$-transversal $W \subseteq V(G)$ with $|W| \leq k$, i.e., a set W such that $F_i \cap W \neq \emptyset$ for every $i \in [q]$?

We emphasize here that the collection $\mathcal{F}$ is implicitly defined in a problem specific-way and we *do not* assume that it is given explicitly in the input, in fact, it is possible that $\mathcal{F}$ is exponentially large. For example, in the DIRECTED MULTIWAY CUT problem we take T as the set of terminals and $\mathcal{F}$ as the set of all walks between different terminals; note that $\mathcal{F}$ is clearly T-connected. In the compression version of SUBSET-DFVS, we take T as the solution that we want to compress and $\mathcal{F}$ as the set of all cycles containing a vertex of S; again, $\mathcal{F}$ is T-connected, since if T is a solution, then every cycle containing a vertex of S goes through T.

We define the "shadow" of a solution X as those vertices that are disconnected from T (in either direction) after the removal of X. A common idea in [35, 11] is to ensure first that there is a solution whose shadow is empty, as finding such a shadowless solution can be a significantly easier task. Our generic framework shows that for the $\mathcal{F}$-transversal problems defined above, we can invoke the random sampling of important separators technique and obtain a set which is disjoint from a minimum solution and covers its shadow. What we do with this set, however, is problem specific. Typically, given such a set, we can use (some problem-specific variant of) the "torso operation" to find an equivalent instance that has a shadowless solution. Therefore, we can focus on the simpler

task of finding a shadowless solution; or more precisely, finding any solution under the guarantee that a shadowless solution exists. We believe our framework will provide a useful opening step in the design of FPT algorithms for other transversal and cut problems on directed graphs.

In the case of undirected MULTICUT [35], if there was a shadowless solution, then the problem could be reduced to an FPT problem called ALMOST 2SAT [32, 42]. In the case of DIRECTED MULTIWAY CUT [11], if there was a solution whose shadow is empty, then the problem could be reduced to the undirected version, which was known to be FPT [7, 14, 34]. For SUBSET-DFVS, the situation turns out to be a bit more complicated. As mentioned above, we first use the technique of iterative compression to reduce the problem to an instance where we are given a solution T and we want to find a disjoint solution of size at most k. We define the "shadows" with respect to the solution T that we want to compress, whereas in [11], the shadows were defined with respect to the terminal set T. The "torso" operation we define in this paper is specific to the SUBSET-DFVS problem, as it takes into account the set S and modifies it accordingly. Furthermore, even after ensuring that there is a solution T' whose shadow is empty, we are not done unlike in [11]. We then analyze the structure of the graph $G \setminus T'$ and focus on the last strongly connected component in the topological ordering of this graph, i.e., the strongly connected component which can only have incoming edges from other strongly connected components. We would like to find the subset of T' that separates this component from the rest of the graph. In most cases, a pushing argument can be used to argue that this subset of T' is an important separator, and hence we can branch on removing an important separator from the graph. However, due to the way the set S interacts with the solution T', there is a small number of vertices that behave in a special way. We need surprisingly complex arguments to handle these special vertices.

The paper is organized as follows. Section 2 introduces notation and the preliminary steps of the algorithm, including iterative compression. Section 3 presents the general result on covering shadows of $\mathcal{F}$-transversals. The remaining sections are specific to the SUBSET-DFVS problem: they discuss how to use the techniques of Section 3 to reduce the problem to instances where the existence of shadowless solutions is guaranteed (Section 4) and how to find a solution under the guarantee that a shadowless solution exists (Section 5); the full algorithm is summarized in Section 6. Finally Section 7 concludes the paper.

2   Preliminaries

Observe that a directed graph contains no cycles if and only if it contains no closed walks; moreover, there is a cycle going through S if and only there is a closed walk going through S. For this reason, throughout the paper we use the term closed walks instead of cycles, since it is sometimes easier to show the existence of a closed walk and avoid discussion whether it is a simple cycle or not. A feedback vertex set is a set of vertices that hits all the closed walks of the graph.

Definition 2.1. (feedback vertex set) *Let G be a directed graph. A set $T \subseteq V(G)$ is a* feedback vertex set *of G if $G \setminus T$ does not contain any closed walks.*

This gives rise to the DIRECTED FEEDBACK VERTEX SET (DFVS) problem where we are given a directed graph G and we want to find if G has a feedback vertex set of size at most k. The DFVS problem was shown to be FPT by Chen et al. [8], answering a long-standing open problem in the parameterized complexity community.

In this paper, we consider a generalization of the DFVS problem where given a set $S \subseteq V(G)$, we ask if there exists a vertex set of size $\leq k$ that hits all closed walks passing through S.

> **Subset Directed Feedback Vertex Set (Subset-DFVS)**
> *Input*: A directed graph G, a set $S \subseteq V(G)$, and a positive integer k.
> *Parameter*: k
> *Question*: Does there exist a set $T \subseteq V(G)$ with $|T| \leq k$ such that $G \setminus T$ has no closed walk containing a vertex of S?

It is easy to see that DFVS is a special case of SUBSET-DFVS obtained by setting $S = V(G)$. We also define a variant of SUBSET-DFVS where the set S is a subset of edges. In this variant, we have destroy the following type of closed walks:

Definition 2.2. *(S-closed-walk) Let G be a directed graph and $S \subseteq E(G)$. A closed walk (starting and ending at same vertex) C in G is said to be a S-closed-walk if it contains an edge from S.*

> **Edge Subset Directed Feedback Vertex Set (Edge-Subset-DFVS)**
> *Input* : A directed graph G, a set $S \subseteq E(G)$, and a positive integer k.
> *Parameter* : k
> *Question* : Does there exist a set $T \subseteq V(G)$ with $|T| \leq k$ such that $G \setminus T$ has no S-closed-walks?

The above two problems can be shown to be equivalent as follows. If (G, S, k) is an instance of SUBSET-DFVS we create an instance (G, S', k) of EDGE-SUBSET-DFVS by taking S' as the set of edges incident to any vertex of S. Then any closed walk passing through a vertex of S must pass through an edge of S', and conversely any closed walk passing through an edge of S' must contain a vertex from S.

On the other hand, given an instance (G, S', k) of EDGE-SUBSET-DFVS we create an instance (G', S, k) of SUBSET-DFVS where G' is obtained from G by the following modification: For every directed edge $(u, v) \in E(G)$ we add a new vertex x_{uv} and path $u \to x_{uv} \to v$ of length 2. We set $S = \{x_e : e \in S'\}$. Then any closed walk in G passing through an edge of S' corresponds to a closed-walk in G' which must pass through a vertex of S, and conversely any closed walk in G' passing through a vertex of S can be easily converted to a closed walk in G passing through an edge of S'. Both the reductions work in polynomial time and do not change the parameter. Therefore, in the rest of the paper we concentrate on solving the EDGE SUBSET DIRECTED FEEDBACK VERTEX SET problem and we shall refer to both the above problems as SUBSET-DFVS.

2.1 Iterative Compression

The first step of our algorithm is to use the technique of *iterative compression* introduced by Reed et al. [43]. It has been used to obtain faster FPT algorithms for various problems [6, 8, 16, 22, 23, 24, 35, 42]. We transform the SUBSET-DFVS problem into the following problem:

> **Subset-DFVS Compression**
> *Input*: A directed graph G, a set $S \subseteq E(G)$, a positive integer k, and a set $T \subseteq V$ such that $G \setminus T$ has no S-closed-walks.
> *Parameter*: $k + |T|$
> *Question*: Does there exist a set $T' \subseteq V(G)$ with $|T'| \leq k$ such that $G \setminus T'$ has no S-closed-walks?

Lemma 2.3. *(power of iterative compression)* SUBSET-DFVS *can be solved by $O(n)$ calls to an algorithm for the* SUBSET-DFVS COMPRESSION *problem with $|T| \leq k + 1$.*

Proof. Let $V(G) = \{v_1, \ldots, v_n\}$ and for $i \in [n]$ let $V_i = \{v_1, \ldots v_i\}$. We construct a sequence of subsets $X_i \subseteq V_i$, such that X_i is a solution for $G[V_i]$. Clearly, $X_1 = \emptyset$ is a solution for $G[V_1]$. Observe that if X_i is a solution for $G[V_i]$, then $X_i \cup \{v_{i+1}\}$ is a solution for $G[V_{i+1}]$. Therefore, for each $i \in [n-1]$, we set $T = X_i \cup \{v_{i+1}\}$ and use, as a blackbox, an algorithm for SUBSET-DFVS COMPRESSION, to construct a set X_{i+1} that is a solution of size at most k for $G[V_{i+1}]$. Note that if there is no solution for $G[V_i]$ for some $i \in [n]$, then there is no solution for the whole graph G and moreover, since $V_n = V(G)$, if all the calls to the reduction problem are successful, then X_n is a solution for the graph G. $\square$

Now we transform the SUBSET-DFVS COMPRESSION problem into the following problem whose only difference is that the subset feedback vertex set in the output must be disjoint from the one in the input:

Disjoint Subset-DFVS Compression
Input: A directed graph G, a set $S \subseteq E(G)$, a positive integer k, and a set $T \subseteq V$ such that $G \setminus T$ has no S-closed-walks.
Parameter: $k + |T|$
Question: Does there exist a set $T' \subseteq V(G)$ with $|T'| \leq k$ such that $T \cap T' = \emptyset$ and $G \setminus T'$ has no S-closed-walks?

Lemma 2.4. *(adding disjointness)* SUBSET-DFVS COMPRESSION *can be solved by* $O(2^{|T|})$ *calls to an algorithm for the* DISJOINT SUBSET-DFVS COMPRESSION *problem.*

Proof. Given an instance $I = (G, S, T, k)$ of SUBSET-DFVS COMPRESSION we guess the intersection X of T and the subset feedback vertex set T' in the output. We have at most $2^{|T|}$ choices for X. Then for each guess for X, we solve the DISJOINT SUBSET-DFVS COMPRESSION problem for the instance $I_X = (G \setminus X, S, T \setminus X, k - |X|)$. It is easy to see that if T' is a solution for instance I of SUBSET-DFVS COMPRESSION, then $T' \setminus X$ is a solution of instance I_X of DISJOINT SUBSET-DFVS COMPRESSION for $X = T' \cap T$. Conversely, if T'' is a solution to some instance I_X, then $T'' \cup X$ is a solution for X. $\square$

From Lemmas 2.3 and 2.4, an FPT algorithm for DISJOINT SUBSET-DFVS COMPRESSION translates into an FPT algorithm for SUBSET-DFVS with an additional blowup factor of $O(2^{|T|}n)$ in the running time.

3 General $\mathcal{F}$-transversal Problems: Covering the Shadow of a Solution

The purpose of this section is to present the "random sampling of important separators" technique developed in [11] for DIRECTED MULTIWAY CUT in a generalized way that applies to SUBSET-DFVS as well. The technique consists of two steps:

1. First find a set Z *small* enough to be disjoint from a solution X (of size $\leq k$) but *large* enough to cover the "shadow" of X.
2. Then define a "torso" operation that uses the set Z to reduce the problem instance in such a way that X becomes a shadowless solution of the reduced instance.

In this section, we define a general family of problems for which Step 1 can be efficiently performed. The general technique to execute Step 1 is very similar to what was done for DIRECTED MULTIWAY CUT [11]. In Section 4, we show how Step 2 can be done for the specific problem of DISJOINT SUBSET-DFVS COMPRESSION. First we start by defining separators and shadows. Following [11], we define separators in a generalized setting where we assume that the graph G is equipped with a subset $V^\infty(G)$ of undeletable vertices and separators by definition have to be disjoint from this set. This extension will be very convenient in the proofs of Section 3.3.

Definition 3.1. (separator) *Let G be a directed graph and $V^\infty(G)$ be the set of distinguished ("undeletable") vertices. Given two disjoint non-empty sets $X, Y \subseteq V$, we call a set $W \subseteq V \setminus (X \cup Y \cup V^\infty)$ an $X - Y$ separator if there is no path from X to Y in $G \setminus W$. A set W is a minimal $X - Y$ separator if no proper subset of W is an $X - Y$ separator.*

Note that here we explicitly define the $X - Y$ separator W to be disjoint from X and Y.

Definition 3.2. (shadows) *Let G be graph and T be a set of terminals. Let $W \subseteq V(G) \setminus V^\infty(G)$ be a subset of vertices.*

1. *The forward shadow $f_{G,T}(W)$ of W (with respect to T) is the set of vertices v such that W is a $T - \{v\}$ separator in G.*
2. *The reverse shadow $r_{G,T}(W)$ of W (with respect to T) is the set of vertices v such that W is a $\{v\} - T$ separator in G.*

The shadow of W (with respect to T) is the union of $f_{G,T}(W)$ and $r_{G,T}(W)$.

That is, we can imagine T as a light source with light spreading on the directed edges. The forward shadow is the set of vertices that remain dark if the set W blocks the light, hiding v from T's sight. In the reverse shadow, we imagine that light is spreading on the edges backwards. We abuse the notation slightly and write $v - T$ separator instead of $\{v\} - T$ separator. We also drop G and T from the subscript if they are clear from the context. Note that W itself is not in the shadow of W (as, by definition, a $T - v$ or $v - T$ separator needs to be disjoint from T and v), that is, W and $f_{G,T}(W) \cup r_{G,T}(W)$ are disjoint. See Figure 1 for an illustration.

Let G be a directed graph and $T \subseteq V(G)$. Let $\mathcal{F} = \{F_1, F_2, \ldots, F_q\}$ be a set of subgraphs of G. We define the following property:

Definition 3.3. (T-connected) *Let $\mathcal{F} = \{F_1, F_2, \ldots, F_q\}$ be a set of subgraphs of G. For a set $T \subseteq V$, we say that $\mathcal{F}$ is T-connected if for every $i \in [q]$, each vertex of the subgraph F_i can reach some vertex of T by a walk completely contained in F_i and is reachable from some vertex of T by a walk completely contained in F_i.*

For a set $\mathcal{F}$ of subgraphs of G, an $\mathcal{F}$-transversal is a set of vertices that intersects each subgraph in $\mathcal{F}$.

Definition 3.4. ($\mathcal{F}$-transversal) *Let $\mathcal{F} = \{F_1, F_2, \ldots, F_q\}$ be a set of subgraphs of G. Then $W \subseteq V(G)$ is said to be an $\mathcal{F}$-transversal if $\forall \, i \in [q]$ we have $F_i \cap W \neq \emptyset$.*

The main result of this section is a randomized algorithm for producing a set that covers the shadow of some $\mathcal{F}$-transversal:

Theorem 3.5. (randomized covering of the shadow) *Let $T \subseteq V(G)$. There is an algorithm RandomSet(G, T, k) that runs in $O^*(4^k)$ time and returns a set $Z \subseteq V(G)$ such that for any set $\mathcal{F}$ of T-connected subgraphs, if there exists an $\mathcal{F}$-transversal of size $\leq k$, then the following holds with probability $2^{-O(k^2)}$: there is an $\mathcal{F}$-transversal X of size $\leq k$ such that*
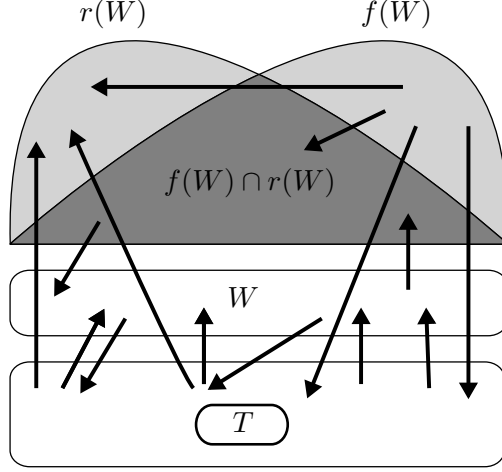
Figure 1: For every vertex $v \in f(W)$, the set W is a $T - v$ separator. For every vertex $w \in r(W)$, the set W is a $w - T$ separator. For every vertex $y \in f(W) \cap r(W)$, the set W is both a $T - y$ and $y - T$ separator. Finally for every $z \in V(G) \setminus [W \cup r(W) \cup f(W) \cup T]$, there are both $z - T$ and $T - z$ paths in the graph $G \setminus W$.

1. $X \cap Z = \emptyset$ and
2. Z covers the shadow of X.

Note that $\mathcal{F}$ is *not* an input of the algorithm described by Theorem 3.5: the set Z constructed in the above theorem works for *every* T-connected set $\mathcal{F}$ of subgraphs. Therefore, issues related to the representation of $\mathcal{F}$ do not arise. Using the theory of splitters, we also prove the following derandomized version of Theorem 3.5:

Theorem 3.6. (deterministic covering of the shadow) *Let $T \subseteq V(G)$. We can construct a set $\{Z_1, Z_2, \ldots, Z_t\}$ with $t = 2^{O(k^2)} \log^2 n$ in time $O^*(2^{O(k^2)})$ such that for any set $\mathcal{F}$ of T-connected subgraphs, if there exists an $\mathcal{F}$-transversal of size $\leq k$, then there is an $\mathcal{F}$-transversal X of size $\leq k$ such that for at least one $i \in [t]$ we have*

1. $X \cap Z_i = \emptyset$ and
2. Z_i covers the shadow of X.

Sections 3.1–3.3 are devoted to the proofs of Theorems 3.5–3.6.

In the DIRECTED MULTIWAY CUT algorithm of Chitnis et al. [11], the set T was the set of terminals and the set $\mathcal{F}$ was the set of all walks from one vertex of T to another vertex of T. Clearly, $\mathcal{F}$ is T-connected: every vertex on a walk from T to T satisfies the reachability conditions. With this interpretation, Theorem 3.6 generalizes Theorem 4.11 of [11] with a better running time. Plugging Theorem 3.6 into the DIRECTED MULTIWAY CUT algorithm of [11] gives an $O^*(2^{O(k^2)})$ time algorithm, proving Theorem 1.2.

In SUBSET-DFVS, the set T is the solution that we want to compress and $\mathcal{F}$ is the set of all S-closed-walks passing through some vertex of T. Again, $\mathcal{F}$ is T-connected: every S-closed-walk goes through T (as T is a solution), hence any vertex on an S-closed-walk is reachable from T, and some vertex of T is reachable from every vertex of the S-closed-walk.

We say that an $\mathcal{F}$-transversal T' is *shadowless* if $f(T') \cup r(T') = \emptyset$. Note that if T' is a shadowless solution, then each vertex of $G \setminus T'$ is reachable from some vertex of T and can reach some vertex

of T. In Section 4, we show that given an instance of DISJOINT SUBSET-DFVS COMPRESSION and a set Z as in Theorem 3.5, we are able to transform the instance using the torso operation in a way that guarantees the existence of the shadowless solution for the reduced instance. In Section 5, we will see how we can make progress in DISJOINT SUBSET-DFVS COMPRESSION if there exists a shadowless solution: we identify a bounded-size set of vertices such that every shadowless solution contains at least one vertex of this set. Therefore, we can branch on including one vertex of this set into the solution.

3.1   Important separators and random sampling

This subsection reviews the notion of important separators and the random sampling technique introduced in [35]. These ideas were later adapted and generalized for directed graphs in [11]. We closely follow [11], but we deviate from it in two ways: we state the results in the framework of $\mathcal{F}$-transversal problems and improve the random selection and its analysis to achieve better running time. Unfortunately, this means that we have to go step-by-step through most of the corresponding arguments of [11]. While some of the statements and proofs are almost the same as in [11], we give a self-contained presentation without relying on earlier work (with the exception of the proof of Lemma 3.8).

3.1.1   Important separators

Marx [34] introduced the concept of *important separators* to deal with the UNDIRECTED MULTIWAY CUT problem. Since then it has been used implicitly or explicitly in [7, 8, 11, 28, 31, 33, 35, 42] in the design of fixed-parameter algorithms. In this section, we define and use this concept in the setting of directed graphs. Roughly speaking, an important separator is a separator of small size that is *maximal* with respect to the set of vertices on one side. Recall that, as in Definition 3.1, the graph G has a set $V^\infty(G)$ of undeletable vertices and an $X - Y$ separator is defined to be disjoint from $X \cup Y \cup V^\infty(G)$.

Definition 3.7. (important separator) *Let G be a directed graph and let $X, Y \subseteq V$ be two disjoint non-empty sets. A minimal $X - Y$ separator W is called an* important $X - Y$ *separator if there is no $X - Y$ separator W' with $|W'| \leq |W|$ and $R^+_{G \setminus W}(X) \subset R^+_{G \setminus W'}(X)$, where $R^+_A(X)$ is the set of vertices reachable from X in the graph A.*

Let X, Y be disjoint sets of vertices of an *undirected graph*. Then for every $k \geq 0$, it is known [7, 34] that there are at most 4^k important $X - Y$ separators of size at most k for any sets X, Y. The next lemma shows that the same bound holds for important separators even in directed graphs.

Lemma 3.8 ([11]). **(number of important separators)** *Let $X, Y \subseteq V(G)$ be disjoint sets in a directed graph G. Then for every $k \geq 0$ there are at most 4^k important $X - Y$ separators of size at most k. Furthermore, we can enumerate all these separators in time $O(4^k \cdot k(|V(G) + |E(G)|))$.*

For ease of notation, we now define the following collection of important separators:

Definition 3.9. *Given a graph G, a set $T \subseteq V(G)$, and an integer k, the set $\mathcal{I}_k$ contains the set $W \subseteq V(G)$ if W is an important $v - T$ separator of size at most k in G for some vertex v in $V(G) \setminus T$.*

Remark 3.10. It follows from Lemma 3.8 that $|\mathcal{I}_k| \leq 4^k \cdot |V(G)|$ and we can enumerate the sets in $\mathcal{I}_k$ in time $O^*(4^k)$.
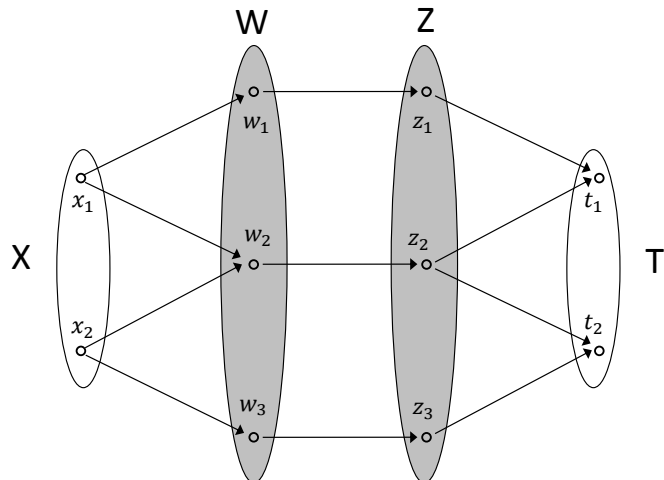
Figure 2: W is a minimal $X - T$ separator, but it is not an important $X - T$ separator as Z satisfies $|Z| = |W|$ and $R^+_{G \setminus W}(X) = X \subset X \cup W = R^+_{G \setminus Z}(X)$. In fact it is easy to check that the only important $X - T$ separator of size 3 is Z. If $k \geq 2$ then the set $\{z_1, z_2\}$ is in $\mathcal{I}_k$, since it is an important $x_1 - T$ separator of size 2. Finally, x_1 belongs to the "exact reverse shadow" of each of the sets $\{w_1, w_2\}, \{w_1, z_2\}, \{w_2, z_1\}$ and $\{z_1, z_2\}$, since they are all minimal $x_1 - T$ separators. However x_1 does not belong to the exact reverse shadow of the set W as it is not a minimal $x_1 - T$ separator.

We now define a special type of shadows which we use later for the random sampling:

Definition 3.11. (exact shadows) *Let G be a directed graph and $T \subseteq V(G)$ a set of terminals. Let $W \subseteq V(G) \setminus V^\infty(G)$ be a subset of vertices. Then for $v \in V(G)$ we say that*

1. *v is in the "exact forward shadow" of W (with respect to T) if W is a minimal $T - v$ separator in G, and*

2. *v is in the "exact reverse shadow" of W (with respect to T) if W is a minimal $v - T$ separator in G.*

We refer the reader to Figure 2 for examples of Definitions 3.7, 3.9 and 3.11. Note that from the two definitions appearing in Defintion 3.11, we will be using only the exact reverse shadow in the paper; the definition of exact forward shadow is given only for completeness. The exact reverse shadow of W is a subset of the reverse shadow of W: it contains a vertex v only if every vertex $w \in S$ is "useful" in separating v from T, i.e., vertex w can be reached from v and T can be reached from w. Similarly for the forward shadow. This slight difference between the shadow and the exact shadow will be crucial in the analysis of the algorithm (Section 3.3).

The weaker version of the random sampling described in Section 3.1.2 (Theorem 3.16) randomly selects members of $\mathcal{I}_k$ and creates a subset by taking the union of the exact reverse shadows of these sets. The following lemma will be used to give an upper bound on the probability that a vertex is covered by the union.

Lemma 3.12. *Let z be any vertex. Then there are at most 4^k members of $\mathcal{I}_k$ that contain z in their exact reverse shadows.*

For the proof of Lemma 3.12, we need to establish first the following:

Lemma 3.13. *If $W \in \mathcal{I}_k$ and v is in the exact reverse shadow of W, then W is an important $v - T$ separator.*

Proof. Let w be the witness that W is in $\mathcal{I}_k$, i.e., W is an important $w - T$ separator in G. Let v be any vertex in the exact reverse shadow of W, which means that W is a minimal $v - T$ separator in G. Suppose that W is not an important $v - T$ separator. Then there exists a $v - T$ separator W' such that $|W'| \leq |W|$ and $R^+_{G \setminus W}(v) \subset R^+_{G \setminus W'}(v)$. We will arrive to a contradiction by showing that $R^+_{G \setminus W}(w) \subset R^+_{G \setminus W'}(w)$, i.e., W is not an important $w - T$ separator.

First, we claim that W' is a $(W \setminus W') - T$ separator. Suppose that there is a path P from some $x \in W \setminus W'$ to T that is disjoint from W'. As W is a minimal $v - T$ separator, there is a path Q from v to x whose internal vertices are disjoint from W. Furthermore, $R^+_{G \setminus W}(v) \subset R^+_{G \setminus W'}(v)$ implies that the internal vertices of Q are disjoint from W' as well. Therefore, concatenating Q and P gives a path from v to T that is disjoint from W', contradicting the fact that W' is a $v - T$ separator.

We show that W' is a $w - T$ separator and its existence contradicts the assumption that W is an important $w - T$ separator. First we show that W' is a $w - T$ separator. Suppose that there is a $w - T$ path P disjoint from W'. Path P has to go through a vertex $y \in W \setminus W'$ (as W is a $w - T$ separator). Thus by the previous claim, the subpath of P from y to T has to contain a vertex of W', a contradiction.

Finally, we show that $R^+_{G \setminus W}(w) \subseteq R^+_{G \setminus W'}(w)$. As $W \neq W'$ and $|W'| \leq |W|$, this will contradict the assumption that W is an important $w - T$ separator. Suppose that there is a vertex $z \in R^+_{G \setminus W}(w) \setminus R^+_{G \setminus W'}(w)$ and consider a $w - z$ path that is fully contained in $R^+_{G \setminus W}(w)$, i.e., disjoint from W. As $z \notin R^+_{G \setminus W'}(w)$, path Q contains a vertex $q \in W' \setminus W$. Since W' is a minimal $v - T$ separator, there is a $v - T$ path that intersects W' only in q. Let P be the subpath of this path from q to T. If P contains a vertex $r \in W$, then the subpath of P from r to T contains no vertex of W' (as $z \neq r$ is the only vertex of W' on P), contradicting our earlier claim that W' is a $(W \setminus W') - T$ separator. Thus P is disjoint from W, and hence the concatenation of the subpath of Q from w to q and the path P is a $w - T$ path disjoint from W, a contradiction. □

Lemma 3.12 easily follows from Lemma 3.13. Let J be a member of $\mathcal{I}_k$ such that z is in the exact reverse shadow of J. By Lemma 3.13, J is an important $z - T$ separator. By Lemma 3.8, there are at most 4^k important $z - T$ separators of size at most k and hence z belongs to at most 4^k exact reverse shadows.

Remark 3.14. It is crucial to distinguish between "reverse shadow" and "exact reverse shadow": Lemma 3.13 (and hence Lemma 3.12) does not remain true if we remove the word "exact." Consider the following example (see Figure 3). Let $a_1, \ldots, a_r$ be vertices such that there is an edge going from every a_i to every vertex of $T = \{t_1, t_2, \ldots, t_k\}$. For every $1 \leq i \leq r$, let b_i be a vertex with an edge going from b_i to a_i. For every $1 \leq i < j \leq r$, let $c_{i,j}$ be a vertex with two edges going from $c_{i,j}$ to a_i and a_j. Then every set $\{a_i, a_j\}$ is in $\mathcal{I}_k$, since it is an important $c_{i,j} - T$ separator; and every set $\{a_i\}$ is in $\mathcal{I}_k$ as well, as it is an important $b_i - T$ separator. Every b_i is in the reverse shadow of $\{a_j, a_i\}$ for $1 \leq i \neq j \leq r$. However, b_i is in the *exact* reverse shadow of exactly one member of $\mathcal{I}_k$, the set $\{a_i\}$.

3.1.2 Random sampling

In this subsection, we describe the technique of random sampling of important separators, which is crucial to the proof of Theorem 3.5. This technique was introduced in [35] and was adapted to
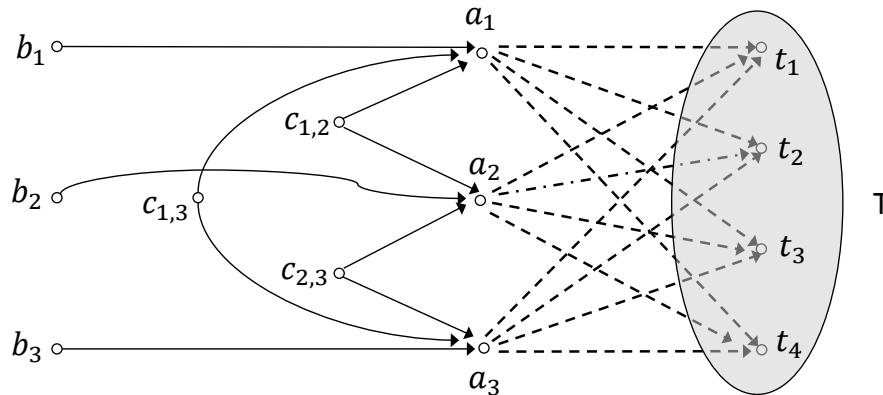
Figure 3: An illustration of Remark 3.14 in the special case when $k = 4$ and $r = 3$.

directed graphs in [11]. We follow it closely and try to present it in a self-contained way that might be useful for future applications. In Section 4, in order to reduce the problem (via the "torso" operation) to a shadowless instance, we need a set Z that has the following property:

> **Property (*)**
> There is an $\mathcal{F}$-transversal T^* of size at most k such that Z covers the shadow of T^*, but Z is disjoint from T^*.

Of course, when we are trying to construct this set Z, we do not know anything about the $\mathcal{F}$-transversals of the instance. In particular we have no way of checking if a given set Z satisfies this property. Nevertheless, we use a randomized procedure that creates a set Z and we give a lower bound on the probability that Z satisfies the requirements. For the construction of this set Z, one can use a very specific probability distribution that was introduced in [35]. This probability distribution is based on randomly selecting "important separators" and taking the union of their shadows. In this paper, we modify the selection of important separators in a way that improves the success probability. The precise description of the randomized procedure and the properties of the distribution it creates is described in Theorems 3.16 and 3.17. Using the theory of splitters we can derandomize the randomized selection into a deterministic algorithm that returns a bounded number of sets such that at least one of them satisfies the required property (Section 3.2).

First we focus on the reverse shadow and try to ensure that (with good probability) Z covers the reverse shadow of T^*. Then in Section 3.3, we argue that, after reversing the orientation of the edges of the graph, a second application of the random selection can be used to cover the forward shadow. Thus in this section, we consider only the reverse shadow of T^*.

Roughly speaking, we want to select a random set Z such that for every every (W, Y) where Y is in the reverse shadow of W, the probability that Z is disjoint from W but contains Y can be bounded from below. We can guarantee such a lower bound only if (W, Y) satisfies two conditions. First, it is not enough that Y is in the shadow of W (or in other words, W is an $Y - T$ separator), but W should contain important separators separating the vertices of Y from T (see Theorems 3.16 and 3.17 for the exact statement). Second, W and Y have to be disjoint, otherwise there is clearly no set covering Y and disjoint from W. In other words, a vertex of W cannot be in the reverse shadow of other vertices of W, which is expressed by the following technical definition:

Definition 3.15. (thin) *Let G be a directed graph. We say that a set $W \subseteq V(G)$ is thin in G if there is no $v \in W$ such that v belongs to the reverse shadow of $W \setminus v$ with respect to T.*

We first give an easy version of the random sampling, which only gives a double exponentially small lower bound on the probability of constructing a set Z with the required properties.

Theorem 3.16. (random sampling) *There is an algorithm* $\mathrm{RandomSet}(G, T, k)$ *that produces a random set* $Z \subseteq V(G) \setminus T$ *in time* $O^*(4^k)$ *such that the following holds. Let* W *be a* thin *set with* $|W| \leq k$, *and let* Y *be a set such that for every* $v \in Y$ *there is an important* $v - T$ *separator* $W' \subseteq W$. *For every such pair* (W, Y), *the probability that the following two events both occur is* $2^{-2^{O(k)}}$:

  1. $W \cap Z = \emptyset$, *and*
  2. $Y \subseteq Z$.

Proof. The algorithm $\mathrm{RandomSet}(G, T, k)$ first enumerates the collection $\mathcal{I}_k$; let $\mathcal{X}$ be the set of all exact reverse shadows of these sets. Note that two different sets in $\mathcal{I}_k$ have different exact reverse shadows: if X is the exact reverse shadow of $J \in \mathcal{I}_k$, then J is exactly the set of vertices not in X and having an inneighbor in X. By Lemma 3.8, the size of $\mathcal{X}$ is $O^*(4^k)$ and can be constructed in time $O^*(4^k)$. Let $\mathcal{X}'$ be the subset of $\mathcal{X}$ where each element from $\mathcal{X}$ occurs with probability $\frac{1}{2}$ independently at random. Let Z be the union of the exact reverse shadows in $\mathcal{X}'$. We claim that the set Z satisfies the requirement of the theorem.

Let us fix a pair (W, Y) as in the statement of the theorem. Let $X_1, X_2, \ldots, X_d \in \mathcal{X}$ be the exact reverse shadows of every member of $\mathcal{I}_k$ that is a subset of W. As $|W| \leq k$, we have $d \leq 2^k$. By the assumption that W is *thin*, we have $X_j \cap W = \emptyset$ for every $j \in [d]$. Now consider the following events:

(E1) $W \cap Z = \emptyset$
(E2) $X_j \subseteq Z$ for every $j \in [d]$

First we show that (E2) implies that $Y \subseteq Z$: $v \in Y$ implies there is an important separator $W' \subseteq W$, i.e., there is some $\ell \in [d]$ such that X_ℓ is the exact reverse shadow of W'. Also note that $v \in X_\ell$ since W' is a minimal (in fact important) $v - T$ separator. Since $X_j \subseteq Z$ for every $j \in [d]$, we have that $v \in Z$. This shows that $Y \subseteq Z$.

Our goal is to show that both events (E1) and (E2) occur with probability $2^{-2^{O(k)}}$. Let $A = \{X_1, X_2, \ldots, X_d\}$ and $B = \{X \in \mathcal{X} \mid X \cap W \neq \emptyset\}$. By Lemma 3.12, each vertex of W is contained in the exact reverse shadows of at most 4^k members of $\mathcal{I}_k$. Thus $|B| \leq |W| \cdot 4^k \leq k \cdot 4^k$. If no exact reverse shadow from B is selected, then event (E1) holds. If every exact reverse shadow from A is selected, then event (E2) holds. Thus the probability that both (E1) and (E2) occur is bounded from below by the probability of the event that every element from A is selected and no element from B is selected. Note that A and B are disjoint: A contains only sets disjoint from W, while B contains only sets intersecting W. Therefore, the two events are independent and the probability that both events occur is at least

$$\left(\frac{1}{2}\right)^{2^k} \left(1 - \frac{1}{2}\right)^{k \cdot 4^k} = 2^{-2^{O(k)}}$$

$\square$

We now give an improved version of the random sampling that gives a stronger lower bound on the success probability than the one guaranteed by Theorem 3.16. Recall that in Theorem 3.16, we randomly selected members of $\mathcal{I}_k$ and took Z as the union of the exact reverse shadows of the selected sets. However, we only had single-exponential upper bounds on both types of exact reverse shadows: number of shadows intersecting W was at most $k \cdot 4^k$ and the number of exact

reverse shadows of every subset of W is at most 2^k. In Theorem 3.17, we take a different view: we randomly select a subset of vertices P and take Z as the union of exact reverse shadows of every subset of $\mathcal{P}$. This will give us a stronger (single exponentially small) lower bound on the probability that the constructed set Z satisfies the required properties.

Theorem 3.17. (improved random sampling) *There is an algorithm* $\mathrm{RandomSet}(G, T, k)$ *that produces a random set $Z \subseteq V(G) \setminus T$ in time $O^*(4^k)$ such that the following holds. Let W be a* thin *set with $|W| \leq k$, and let Y be a set such that for every $v \in Y$ there is an important $v - T$ separator $W' \subseteq W$. For every such pair (W, Y), the probability that the following two events both occur is $2^{-O(k^2)}$:*

1. *$W \cap Z = \emptyset$, and*
2. *$Y \subseteq Z$.*

Proof. The algorithm $\mathrm{RandomSet}(G, T, k)$ picks a subset P of $V(G)$ where each element occurs with probability 4^{-k} uniformly at random. For every $S \in \mathcal{I}_k$ with $S \subseteq P$, let us add the exact reverse shadow of S to $\mathcal{X}'$. Let Z be the union of the exact reverse shadows in $\mathcal{X}'$. We claim that the set Z satisfies the requirement of the theorem.

Fix a pair (W, Y) as in the statement of the theorem. For each $w \in W$, we define

$$\mathcal{L}_w = \{S \mid S \text{ is an important } w - T \text{ separator of size} \leq k\},$$
$$I_w = \bigcup_{S \in \mathcal{L}_w} S,$$
$$I = \bigcup_{w \in W} I_w.$$

Note that a vertex $w \in W$ may have an outneighbor in T, in which case $\mathcal{L}_w$ and I_w are empty. Since $|W| \leq k$ and for each $w \in W$ there are at most 4^k important $w - T$ separators of size at most k, we have $|I_w| \leq k \cdot 4^k$. Since $|W| \leq k$, we have $|I| \leq k^2 \cdot 4^k$.

Let $\mathcal{X}$ be the set of exact reverse shadows of every set $S \in \mathcal{I}_k$. Let $X_1, X_2, \ldots, X_d \in \mathcal{X}$ be the exact reverse shadows of every $S \in \mathcal{I}_k$ with $S \subseteq W$. Let $A = \{X_1, X_2, \ldots, X_d\}$ and $B = \{X \in \mathcal{X} \mid X \cap W \neq \emptyset\}$. Now consider the following events:

(E1) $W \cap Z = \emptyset$
(E2) $X_j \subseteq Z$ for every $j \in [d]$

First we show that (E2) implies that $Y \subseteq Z$: $v \in Y$ implies there is an important separator $W' \subseteq W$, i.e., there is some $\ell \in [d]$ such that X_ℓ is the exact reverse shadow of W'. Also note that $v \in X_\ell$ since W' is a minimal (in fact important) $v - T$ separator. Since $X_j \subseteq Z$ for every $j \in [d]$, we have that $v \in Z$. This shows that $Y \subseteq Z$.

Our goal is to show that both events (E1) and (E2) occur with probability $2^{-O(k^2)}$. If every vertex from W is selected in P, then every reverse shadow from A is selected into $\mathcal{X}'$ and event (E2) holds. We claim that if no vertex from $I \setminus W$ is selected in P, then no exact reverse shadow from B is selected into $\mathcal{X}'$ and hence event $(E1)$ will also hold. Suppose to the contrary that an exact reverse shadow $X \in B$ was selected into $\mathcal{X}'$; by the definition of B, there is a vertex $w \in X \cap W$. Let $J \in \mathcal{I}_k$ be the set whose exact reverse shadow is X, which implies by Lemma 3.13 that $J \in \mathcal{L}_w$ and $J \subseteq I_w \subseteq I$. If $J \setminus W \neq \emptyset$, then the assumption that no vertex of $I \setminus W$ was selected into P condtradicts the fact that X was selected into $\mathcal{X}'$. Suppose therefore that $J \subseteq W$ holds. Since X is the exact reverse shadow of J, we know that J is a minimal $X - T$ separator. But $J \subseteq W$

14

implies that $W \setminus X$ is also an $X - T$ separator, i.e., $W \cap X$ lies in the reverse shadow of $W \setminus X$. This contradicts the fact that W is a thin set (see Definition 3.15).

Thus the probability that both the events (E1) and (E2) occur is bounded from below by the probability of the event that every vertex from W is selected in P and no vertex from $I \setminus W$ is selected in P. Note that the sets W and $I \setminus W$ are clearly disjoint. Therefore, the two events are independent and the probability that both events occur is at least

$$(4^{-k})^k (1 - 4^{-k})^{k^2 \cdot 4^k} \geq 4^{-k^2} \cdot e^{-2k^2} = 2^{-O(k^2)}$$

where we used the inequalities that $1 + x \geq e^{\frac{x}{1+x}}$ for every $x > -1$ and $1 - 4^{-k} \geq \frac{1}{2}$ for every $k \geq 1$. $\qquad \square$

3.2 Derandomization

We now derandomize the process of choosing exact reverse shadows in Theorem 3.17 using the technique of *splitters*. An (n, r, r^2)-splitter is a family of functions from $[n] \to [r^2]$ such that for every $M \subseteq [n]$ with $|M| = r$, at least one of the functions in the family is injective on M. Naor et al. [37] give an explicit construction of an (n, r, r^2)-splitter of size $O(r^6 \log r \log n)$ in time $\text{poly}(n, r)$.

Theorem 3.18. (deterministic sampling) *There is an algorithm* DeterministicSets(G, T, k) *that produces* $t = 2^{O(k^2)} \log |V(G)|$ *subsets of* $Z_1, \ldots, Z_t$ *of* $V(G) \setminus T$ *in time* $O^*(2^{O(k^2)})$ *such that the following holds. Let W be a* thin *set with* $|W| \leq k$, *and let Y be a set such that for every* $v \in Y$ *there is an important* $v - T$ *separator* $W' \subseteq W$. *For every such pair* (W, Y), *there is at least one* $1 \leq i \leq t$ *with*

1. $W \cap Z = \emptyset$, *and*
2. $Y \subseteq Z$.

Proof. In the proof of Theorem 3.17, a random subset P of a universe $V(G)$ of size n is selected. We argued that if every vertex from W is selected in P and no element from $I \setminus W$ is selected, then both the events (E1) and (E2) occur. Instead of selecting a random subset P, we will construct several subsets such that at least one of them will contain every vertex in W and no vertex from $I \setminus W$. Let $n = |V(G)|$, $a = |W| \leq k$, and $b = |I \setminus W| \leq k^2 \cdot 4^k$. Each subset is defined by a pair (h, H), where h is a function in an $(n, a+b, (a+b)^2)$-splitter family and H is a subset of $[(a+b)^2]$ of size a (there are $\binom{(a+b)^2}{a} = \binom{(k+k^2 \cdot 4^k)^2}{k} = 2^{O(k^2)}$ such sets H). For a particular choice of h and H, we select those vertices $v \in V(G)$ into P for which $h(v) \in H$. The size of the splitter family is $O\left((a+b)^6 \log(a+b) \log(n)\right) = 2^{O(k)} \log n$ and the number of possibilities for H is $2^{O(k^2)}$. Therefore, we construct $2^{O(k^2)} \log n$ subsets of $V(G)$. The total time taken for constructing these subsets is $\text{poly}(n, a+b) = \text{poly}(n, 4^k)$.

By the definition of the splitter, there is a function h that is injective on W, and there is a subset H such that $h(v) \in H$ for every set $v \in W$ and $h(y) \notin H$ for every $y \in I \setminus W$. For such an h and H, the selection will ensure that (E1) and (E2) hold. Thus at least one of the constructed subsets has the required properties, which is what we had to show. $\qquad \square$

3.3 Proof of Theorem 3.5: The Covering Algorithm

To prove Theorem 3.5, we show that Algorithm 1 gives a set Z satisfying the properties of Theorem 3.5. Due to the delicate way separators and shadows behave in directed graphs, we construct the set Z in two phases, calling the function RandomSet of Section 3.1 twice and taking Z to be

---

Algorithm 1: COVERING (randomized version)

Input: A directed graph G_1, integer k.

Output: A set Z.

1: Let $Z_1 = \text{RandomSet}(G_1, T, k)$.
2: Let G_2 be obtained from G_1 by reversing the orientation of every edge and adding every vertex of Z_1 to V^∞.
3: Let $Z_2 = \text{RandomSet}(G_2, T, k)$.
4: Let $Z = Z_1 \cup Z_2$.

---

the union of the two outputs. For consistency of notation, we denote the input graph by G_1. Let Z_1 be the output of the first call of the function RandomSet, i.e., $Z_1 = \text{RandomSet}(G_1, T, k)$. We build a new graph G_2 from G_1 by reversing the orientation of every edge and adding every vertex of Z_1 to V^∞. Since the structure of the graph G_2 depends on the set Z_1, the distribution of the second random sampling depends on the result Z_1 of the first random sampling. This means that we cannot make the two calls in parallel. Our aim is to show that there is a transversal T^* such that we can give a lower bound on the probability that Z_1 covers $r_{G_1,T}(T^*)$ and Z_2 covers $f_{G_1,T}(T^*)$.

To prove the existence of the required transversal T^*, we need the following definition:

Definition 3.19. (shadow-maximal transversal) *An $\mathcal{F}$-transversal W is* minimum *if there is no $\mathcal{F}$-transversal of size less than $|W|$. A minimum $\mathcal{F}$-transversal W is called* shadow-maximal *if $r_{G_1,T}(W) \cup f_{G_1,T}(W) \cup W$ is inclusion-wise maximal among all minimum $\mathcal{F}$-transversals.*

For the rest of the proof, let us fix T^* to be a shadow-maximal $\mathcal{F}$-transversal such that $|r_{G_1,T}(T^*)|$ is maximum possible among all shadow-maximal $\mathcal{F}$-transversals. We bound the probability that $Z \cap T^* = \emptyset$ and $r_{G_1,T}(T^*) \cup f_{G_1,T}(T^*) \subseteq Z$. More precisely, we bound the probability that all of the following four events occur:

1. $Z_1 \cap T^* = \emptyset$,
2. $r_{G_1,T}(T^*) \subseteq Z_1$,
3. $Z_2 \cap T^* = \emptyset$, and
4. $f_{G_1,T}(T^*) \subseteq Z_2$.

That is, the first random selection takes care of the reverse shadow, the second takes care of the forward shadow, and none of Z_1 or Z_2 hits T^*. Note that it is somewhat counterintuitive that we choose a T^* for which the shadow is large: intuitively, it seems that the larger the shadow is, the less likely that it is fully covered by Z. However, we need this maximality property in order to bound the probability that $Z \cap T^* = \emptyset$.

We want to invoke Theorem 3.17 to bound the probability that Z_1 covers $Y = r_{G_1,T}(T^*)$ and $Z_1 \cap T^* = \emptyset$. First, we need to ensure that T^* is a *thin* set, but this follows easily from the fact that T^* is a minimum $\mathcal{F}$-transversal:

Lemma 3.20. *If W is a minimum $\mathcal{F}$-transversal for some T-connected $\mathcal{F}$, then no $v \in W$ is in the reverse shadow of some $W' \subseteq W \setminus v$.*

Proof. Suppose to the contrary that there is a vertex $v \in W$ such that $v \in r(W')$ for some $W' \subseteq W \setminus v$. Then we claim that $W \setminus v$ is also an $\mathcal{F}$-transversal, contradicting the minimality of W. Let $\mathcal{F} = \{F_1, F_2, \ldots, F_q\}$ and suppose that there is a $i \in [q]$ such that $F_i \cap W = \{v\}$. As $\mathcal{F}$ is T-connected, there is a $v \to T$ walk P in F_i. But $P \cap W = \{v\}$ implies that there is a $v \to T$ walk in $G \setminus (W \setminus v)$, i.e., v cannot belong to the reverse shadow of any $W' \subseteq W \setminus v$. $\qquad \square$

More importantly, if we want to use Theorem 3.17 with $Y = r_{G_1,T}(T^*)$, then we have to make sure that for every vertex v of $r_{G_1,T}(T^*)$, there is an important $v - T$ separator that is a subset of T^*. The "pushing argument" of Lemma 3.21 shows that if this is not true for some v, then we can modify the $\mathcal{F}$-transversal in a way that increases the size of the reverse shadow. The extremal choice of T^* ensures that no such modification is possible, thus T^* contains an important $v - T$ separator for every v.

Lemma 3.21. (pushing) *Let W be an $\mathcal{F}$-transversal for some T-connected $\mathcal{F}$. For every $v \in r(W)$, either there is a $W_1 \subseteq W$ that is an important $v - T$ separator, or there is an $\mathcal{F}$-transversal W' such that*

1. *$|W'| \leq |W|$,*
2. *$r(W) \subset r(W')$,*
3. *$(r(W) \cup f(W) \cup W) \subseteq (r(W') \cup f(W') \cup W')$.*

Proof. Let W_0 be the subset of W reachable from v without going through any other vertices of W. Then W_0 is clearly a $v - T$ separator. Let W_1 be the minimal $v - T$ separator contained in W_0 (we may note that if W is a minimal $\mathcal{F}$-transversal, then we always have $W_1 = W_0$). If W_1 is an important $v - T$ separator, then we are done as W itself contains W_1. Otherwise, there exists an important $v - T$ separator W_1', i.e., $|W_1'| \leq |W_1|$ and $R_{G \setminus W_1}^+(v) \subset R_{G \setminus W_1'}^+(v)$. Now we show that $W' = (W \setminus W_1) \cup W_1'$ is also an $\mathcal{F}$-transversal. Note that $W_1' \subseteq W'$ and $|W'| \leq |W|$.

First we claim that $r(W) \cup (W \setminus W') \subseteq r(W')$. Suppose that there is a walk P from β to T in $G \setminus W'$ for some $\beta \in r(W) \cup (W \setminus W')$. If $\beta \in r(W)$, then walk P has to go through a vertex $\beta' \in W$. As β' is not in W', it has to be in $W \setminus W'$. Therefore, by replacing β with β', we can assume in the following that $\beta \in W \setminus W' \subseteq W_1 \setminus W_1'$. By the minimality of W_1, every vertex of $W_1 \subseteq W_0$ has an incoming edge from some vertex in $R_{G \setminus W}^+(v)$. This means that there is a vertex $\alpha \in R_{G \setminus W}^+(v)$ such that $(\alpha, \beta) \in E(G)$. Since $R_{G \setminus W}^+(v) \subset R_{G \setminus W'}^+(v)$, we have $\alpha \in R_{G \setminus W'}^+(v)$, implying that there is a $v \to \alpha$ walk in $G \setminus W'$. The edge $\alpha \to \beta$ also survives in $G \setminus W'$ as $\alpha \in R_{G \setminus W'}^+(v)$ and $\beta \in W_1 \setminus W_1'$. By assumption, we have a walk in $G \setminus W'$ from β to some $t \in T$. Concatenating the three walks we obtain a $v \to t$ walk in $G \setminus W'$, which contradicts the fact that W' contains an (important) $v - T$ separator W_1'. This proves the claim. Since $W \neq W'$ and $|W| = |W'|$, the set $W_1 \setminus W_1'$ is non-empty. Thus $r(W) \subset r(W')$ follows from the claim $r(W) \cup (W \setminus W') \subseteq r(W')$.

Suppose now that W' is not an $\mathcal{F}$-transversal. Then there is some $i \in [q]$ such that $F_i \cap W' = \emptyset$. As W is an $\mathcal{F}$-transversal, there is some $w \in W \setminus W'$ with $w \in F_i$. As $\mathcal{F}$ is T-connected, there is a $w \to T$ walk in F_i, which gives a $w \to T$ walk in $G \setminus W'$ as $W' \cap F_i = \emptyset$. However, we have $W \setminus W' \subseteq r(W')$ (by the claim in the previous paragraph), a contradiction. Thus W' is also an $\mathcal{F}$-transversal.

Finally, we show that $r(W) \cup f(W) \cup W \subseteq r(W') \cup f(W') \cup W'$. We know that $r(W) \cup (W \setminus W') \subseteq r(W')$. Thus it is sufficient to consider a vertex $v \in f(W) \setminus r(W)$. Suppose that $v \notin f(W')$ and $v \notin r(W')$: there are walks P_1 and P_2 in $G \setminus W'$, going from T to v and from v to T, respectively. As $v \in f(W)$, walk P_1 intersects W, i.e., it goes through a vertex of $\beta \in W \setminus W' \subseteq r(W')$. However, concatenating the subwalk of P_1 from β to v and the walk P_2 gives a walk from $\beta \in r(W')$ to T in $G \setminus W'$, a contradiction. $\square$

Note that if W is a shadow-maximal $\mathcal{F}$-transversal, then the $\mathcal{F}$-transversal W' in Lemma 3.21 is also a minimum $\mathcal{F}$-transversal and shadow-maximal. Therefore, by the extremal choice of T^*, applying Lemma 3.21 on T^* cannot produce a shadow-maximal $\mathcal{F}$-transversal T' with $r_{G_1,T}(T^*) \subset r_{G_1,T}(T')$, and hence T^* contains an important $v - T$ separator for every $v \in r_{G_1,T}(T^*)$. Thus by Theorem 3.17 for $Y = r_{G_1,T}(T^*)$, we get:

Lemma 3.22. *With probability at least $2^{-O(k^2)}$, both $r_{G_1,T}(T^*) \subseteq Z_1$ and $Z_1 \cap T^* = \emptyset$ occur.*

In the following, we assume that the events in Lemma 3.22 occur. Our next goal is to bound the probability that Z_2 covers $f_{G_1,T}(T^*)$. Let us define a collection $\mathcal{F}'$ of subgraphs of G_2 as follows: for every subgraph $F \in \mathcal{F}$ of G_1, let us add to $\mathcal{F}'$ the corresponding subgraph F' of G_2, i.e., F' is the same as F with every edge reversed. Note that $\mathcal{F}'$ is T-connected in G_2: the definition of T-connected is symmetric with respect to the orientation of the edges. Moreover, T^* is an $\mathcal{F}'$-transversal in G_2: the vertices in T^* remained finite (as $Z_1 \cap T^* = \emptyset$ by Lemma 3.22), and reversing the orientation of the edges does not change the fact that T^* is a transversal. Set T^* is also shadow-maximal as an $\mathcal{F}'$-transversal in G_2: Definition 3.19 is insensitive to reversing the orientation of the edges and adding some vertices to V^∞ can only decrease the set of potential transversals. Furthermore, the forward shadow of T^* in G_2 is same as the reverse shadow of T^* in G_1, that is, $f_{G_2,T}(T^*) = r_{G_1,T}(T^*)$. Therefore, assuming that the events in Lemma 3.22 occur, every vertex of $f_{G_2,T}(T^*)$ is in V^∞ in G_2. We show that now it holds that T^* contains an important $v - T$ separator in G_2 for every $v \in r_{G_2,T}(T^*) = f_{G_1,T}(T^*)$:

Lemma 3.23. *If W is a shadow-maximal $\mathcal{F}$-transversal for some T-connected $\mathcal{F}$ and every vertex of $f(W)$ belongs to V^∞, then W contains an important $v - T$ separator for every $v \in r(W)$.*

Proof. Suppose to the contrary that there exists $v \in r(W)$ such that W does not contain an important $v - T$ separator. Then by Lemma 3.21, there is a another shadow-maximal $\mathcal{F}$-transversal W'. As W is shadow-maximal, it follows that $r(W) \cup f(W) \cup W = r(W') \cup f(W') \cup W'$. Therefore, the nonempty set $W' \setminus W$ is fully contained in $r(W) \cup f(W) \cup W$. However, it cannot contain any vertex of $f(W)$ (as they are infinite by assumption) and cannot contain any vertex of $r(W)$ (as $r(W) \subset r(W')$), a contradiction. $\square$

Recall that T^* is a shadow-maximal $\mathcal{F}'$-transversal in G_2. In particular, T^* is a minimal $\mathcal{F}'$-transversal in G_2, hence Lemma 3.20 implies that T^* is thin in G_2 also. Thus Theorem 3.17 can be used again (this time with $Y = r_{G_2,T}(T^*)$) to bound the probability that $r_{G_2,T}(T^*) \subseteq Z_2$ and $Z_2 \cap T^* = \emptyset$. As the reverse shadow $r_{G_2,T}(T^*)$ in G_2 is the same as the forward shadow $f_{G_1,T}(T^*)$ in G, we can state the following:

Lemma 3.24. *Assuming the events in Lemma 3.22 occur, with probability at least $2^{-O(k^2)}$ both $f_{G_1,T}(T^*) \subseteq Z_2$ and $Z_2 \cap T^* = \emptyset$ occur.*

Therefore, with probability $(2^{-O(k^2)})^2$, the set $Z_1 \cup Z_2$ covers $f_{G_1,T}(T^*) \cup r_{G_1,T}(T^*)$ and it is disjoint from T^*. This completes the proof of Theorem 3.5.

Finally, to prove Theorem 3.6, the derandomized version of Theorem 3.6, we use the deterministic variant DeterministicSets(G, T, k) of the function RandomSet(G, T, k) that, instead of returning a random set Z, returns a deterministic set $Z_1, \ldots, Z_t$ of $t = 2^{O(k^2)} \log n$ sets in poly$(n, 4^k)$ time (Theorem 3.18). Therefore, in Steps 1 and 3 of Algorithm 1, we can replace RandomSet with this deterministic variant DeterministicSets, and branch on the choice of one Z_i from the returned sets. By the properties of the deterministic algorithm, if I is a yes-instance, then Z has Property (*) in at least one of the $2^{O(k^2)} \log^2 n$ branches. The branching increases the running time only by a factor of $(O^*(2^{O(k^2)}))^2$ and therefore the total running time is $O^*(2^{O(k^2)})$. This completes the proof of Theorem 3.6.

4 Disjoint Subset-DFVS Compression: Reduction to Shadowless Solutions

We use the algorithm of Theorem 3.6 to construct a set Z of vertices that we want to get rid of. The second ingredient of our algorithm is an operation that removes a set of vertices without making the problem any easier. This transformation can be conveniently described using the operation of taking the *torso* of a graph. From this point onwards in the paper, we do not follow [11]. In particular, the *torso* operation is problem-specific. For DISJOINT SUBSET-DFVS COMPRESSION, we define it as follows:

Definition 4.1. (torso) *Let (G, S, T, k) be an instance of* DISJOINT SUBSET-DFVS COMPRESSION *and $C \subseteq V(G)$. Then* torso(G, C, S) *is a pair (G', S') defined as follows:*

- *G' has vertex set C and there is (directed) edge (a, b) in G' if there is an $a \to b$ walk in G whose internal vertices are not in C,*
- *S' contains those edges of S whose endpoints are both in C; furthermore, we add the edge (a, b) to S' if there is an $a \to b$ walk in G that contains an edge from S and whose internal vertices are not in C.*

In particular, if $a, b \in C$ and (a, b) is a directed edge of G and torso$(G, C, S) = (G', S')$, then G' contains (a, b) as well. Thus G' is a supergraph of the subgraph of G induced by C. Figure 4 illustrates the definition of torso with an example.

The following easy statement was proved in [11]: the torso operation preserves whether a set $W \subseteq C$ is a separator.

Lemma 4.2 ([11]). **(torso preserves separation)** *Let G be a directed graph and $C \subseteq V(G)$. Let $(G', \emptyset) = $ torso$(G, C, \emptyset)$ and $W \subseteq C$. For $a, b \in C \setminus W$, the graph $G \setminus W$ has an $a \to b$ path if and only if $G' \setminus W$ has an $a \to b$ path.*

We need a very similar statement here: the torso operation preserves whether a set $W \subseteq C$ hits every S-closed-walk.

Lemma 4.3. (torso preserves S-closed-walks) *Let G be a directed graph with $C \subseteq V(G)$ and $S \subseteq E(G)$. Let $(G', S') = $ torso(G, C, S), $v \in C$, and $W \subseteq C$. Then $G \setminus W$ has an S-closed-walk passing through v if and only if $G' \setminus W$ has an S'-closed-walk passing through v.*

Proof. Let P be an S-closed-walk in $G \setminus W$ passing through v. If P is fully contained in C, then P also appears in $G' \setminus W$. Otherwise, P contains vertices from both C and $V(G) \setminus C$. Let u, w be two vertices of C such that every vertex of P between u and w is from $V(G) \setminus C$. Then, by definition of torso, there is an edge (u, w) in G'. Using such edges, we can modify P to obtain another closed walk say P' passing through v that lies completely in G' but avoids W. Note that since P is a S-closed-walk, at least one of the edges on some $u \to w$ walk that we short-circuited above must have been from S and by Definition 4.1 we would have put the edge (u, w) edge into S', which makes P' an S'-closed-walk in G'.

Conversely, suppose that P' is an S'-closed-walk passing through a vertex v in G' and it avoids $W \subseteq C$. If P' uses an edge $(u, w) \notin E(G)$, then this means that there is a $u \to w$ walk P_{uw} whose internal vertices are not in C. Using such walks, we modify P' to get a closed walk P passing through v that only uses edges from G, i.e., P is a closed walk in $G \setminus W$. It remains to show that P is an S-closed-walk: since P' is an S'-closed-walk, either some edge of P' was originally in S or there exist some $a, b \in P'$ such that there is a $a \to b$ walk does not contain any vertex from C and some edge on this walk was originally in S. $\qquad\square$

The graph G

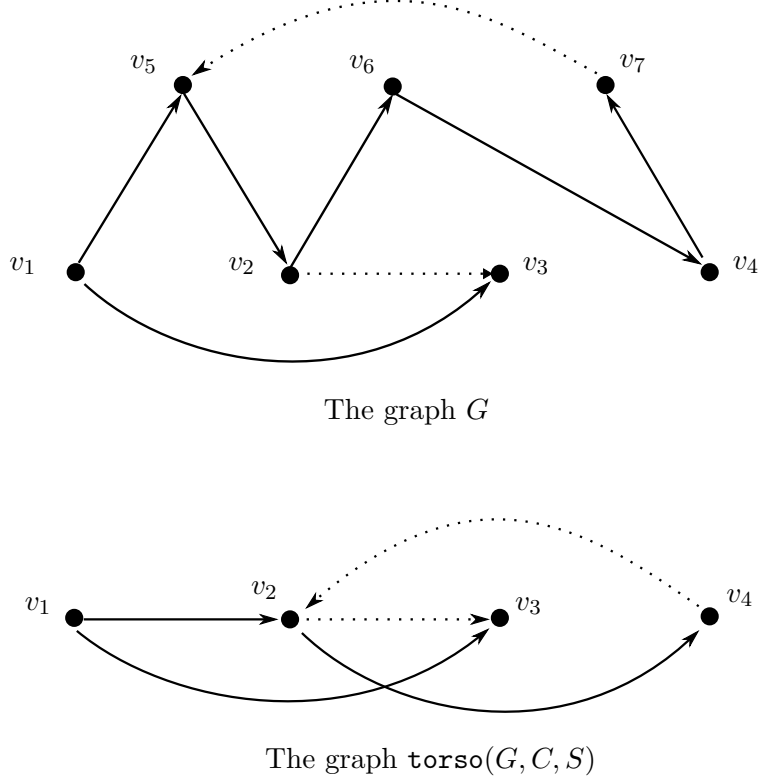The graph $\texttt{torso}(G, C, S)$

Figure 4: In the top graph G we have $C = \{v_1, v_2, v_3, v_4\}$. The edges in S are given by the dotted lines. In the bottom graph we show the graph $\texttt{torso}(G, C, S)$. All edges from $G[C]$ appear in this graph. In addition, we also add the edges $(v_1, v_2), (v_2, v_4)$ and (v_4, v_2). The edge $(v_2, v_3) \in G[C] \cap S$ appears in S'. In addition, we also add the edge (v_4, v_2) to S' since the $v_4 \to v_7 \to v_5 \to v_2$ path in G has an edge $(v_7, v_5) \in S$.

If we want to remove a set Z of vertices, then we create a new instance by taking the torso on the *complement* of Z:

Definition 4.4. *Let $I = (G, S, T, k)$ be an instance of* Disjoint Subset-DFVS Compression *and $Z \subseteq V(G) \setminus T$. The reduced instance $I/Z = (G', S', T, k)$ is obtained by setting $(G', S') = \texttt{torso}(G, V(G) \setminus Z, S)$.*

The following lemma states that the operation of taking the torso does not make the Disjoint Subset-DFVS Compression problem easier for any $Z \subseteq V(G) \setminus T$ in the sense that any solution of the reduced instance I/Z is a solution of the original instance I. Moreover, if we perform the torso operation for a Z that is large enough to cover the shadow of some solution T^* and also small enough to be disjoint from T^*, then T^* becomes a shadowless solution for the reduced instance I/Z.

Lemma 4.5. (creating a shadowless instance) *Let $I = (G, S, T, k)$ be an instance of* Disjoint Subset-DFVS Compression *and $Z \subseteq V(G) \setminus T$.*

20

1. *If I is a no-instance, then the reduced instance I/Z is also a no-instance.*
2. *If I has solution T' with $f_{G,T}(T') \cup r_{G,T}(T') \subseteq Z$ and $T' \cap Z = \emptyset$, then T' is a shadowless solution of I/Z.*

Proof. Let $C = V(G) \setminus Z$ and $(G', S') = \texttt{torso}(G, C, S)$. To prove the first statement, suppose that $T' \subseteq V(G')$ is a solution for I/Z. We show that T' is also a solution for I. Suppose to the contrary that $G \setminus T'$ has an S-closed-walk, which has to pass through some vertex $v \in T$ (since $G \setminus T$ has no S-closed-walks). Note that $v \in T$ and $Z \subseteq V(G) \setminus T$ implies $v \in C$. Then by Lemma 4.3, $G' \setminus T'$ also has an S'-closed-walk passing through v contradicting the fact that T' is a solution for I/Z.

For the second statement, let T' be a solution of I with $T' \cap Z = \emptyset$ and $f_{G,T}(T') \cup r_{G,T}(T') \subseteq Z$. We claim T' is a solution of I/Z as well. Suppose to the contrary that $G' \setminus T'$ has an S'-closed-walk passing through some vertex $v \in C$. As $v \in C$, Lemma 4.3 implies $G \setminus T'$ also has an S-closed-walk passing through v, which is a contradiction as T' is a solution of I.

Finally, we show that T' is a shadowless solution, i.e, $r_{G',T}(T') = f_{G',T}(T') = \emptyset$. We only prove $r_{G',T}(T') = \emptyset$: the argument for $f_{G',T}(T') = \emptyset$ is analogous. Assume to the contrary that there exists $w \in r_{G',T}(T')$ (note that we have $w \in V(G')$, i.e., $w \notin Z$). This means that T' is a $w - T$ separator in G', i.e., there is no $w - T$ walk in $G' \setminus T'$. By Lemma 4.2, it follows that there is no $w - T$ walk in $G \setminus T'$ either, i.e., $w \in r_{G,T}(T')$. But $r_{G,T}(T') \subseteq Z$ and therefore we have $w \in Z$, which is a contradiction. $\square$

For every Z_i in the output of Theorem 3.6, we use the torso operation to remove the vertices in Z_i. We prove that this procedure is safe in the following sense:

Lemma 4.6. *Let $I = (G, S, T, k)$ be an instance of* DISJOINT SUBSET-DFVS COMPRESSION. *Let the sets in the output of Theorem 3.6 be $Z_1, Z_2, \ldots, Z_t$. For every $i \in [t]$, let G_i be the reduced instance G/Z_i.*

1. *If I is a no-instance, then G_i is also a no-instance for every $i \in [t]$.*
2. *If I is a yes-instance, then there exists a solution T^* of I which is a shadowless solution of some G_j for some $j \in [t]$.*

Proof. The first claim is easy to see: any solution T' of the reduced instance (G_i, S, T, k) is also a solution of (G, S, T, k) (by Lemma 4.5(1), the torso operation does not make the problem easier by creating new solutions).

By the derandomization of COVERING algorithm, there is a $j \in [t]$ such that Z has the Property (*), i.e., there is a solution T^* of I such that $Z \cap T^* = \emptyset$ and Z covers shadow of T^*. Then Lemma 4.5(2) implies that T^* is a shadowless solution for the instance $G_j = I/Z_j$. $\square$

5 Disjoint Subset-DFVS Compression: Finding a Shadowless Solution

Consider an instance (G, S, T, k) of DISJOINT SUBSET-DFVS COMPRESSION. First, let us assume that we can reach a start point of some edge of S from each vertex of T, since otherwise we can clearly remove such a vertex from the graph (and from the set T) without changing the problem. Next, we branch on all $2^{O(k^2)} \log^2 n$ choices for Z taken from $\{Z_1, Z_2, \ldots, Z_t\}$ (given by Theorem 3.6) and build a reduced instance I/Z for each choice of Z. By Lemma 4.6, if I is a no-instance, then I/Z_j is a no-instance for each $j \in [t]$. If I is a yes-instance, then by Lemma 4.6 there is at least one $i \in [t]$ such that I has a shadowless solution for the reduced instance I/Z_i.
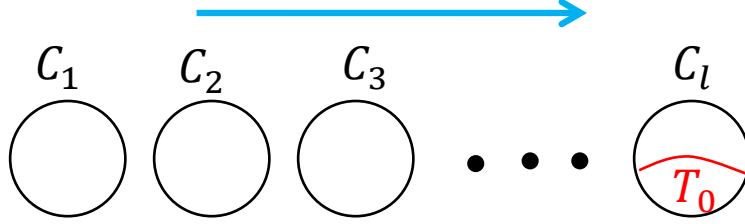
Figure 5: We arrange the strongly connected components of $G \setminus T'$ in a topological order so that the only possible direction of edges between the strongly connected components is as shown by the blue arrow. We will show later that the last component C_ℓ must contain a non-empty subset T_0 of T and further that no edge of S can be present within C_ℓ. This allows us to make some progress as we shall see in Theorem 5.5.

Let us consider the branch where $Z = Z_i$ and let $T' \subseteq V \setminus T$ be a hypothetical shadowless solution for I/Z. We know that each vertex in $G \setminus T'$ can reach some vertex of T and can be reached from a vertex of T. Since T' is a solution for the instance (G, S, T, k) of DISJOINT SUBSET-DFVS COMPRESSION, we know that $G \setminus T'$ does not have any S-closed-walks. Consider a topological ordering $C_1, C_2, \ldots, C_\ell$ of the strongly connected components of $G \setminus T'$, i.e., there can be an edge from C_i to C_j only if $i \leq j$. We illustrate this in Figure 5.

Definition 5.1. (starting/ending points of S) *Let S^- and S^+ be the sets of starting and ending points of edges in S respectively, i.e., $S^- = \{u \mid (u, v) \in S\}$ and $S^+ = \{v \mid (u, v) \in S\}$.*

Lemma 5.2. (properties of C_ℓ) *For a shadowless solution T' for an instance of* DISJOINT SUBSET-DFVS COMPRESSION, *let C_ℓ be the last strongly connected component in the topological ordering of $G \setminus T'$ (refer to Figure 5). Then*

1. *C_ℓ contains a non-empty subset T_0 of T.*
2. *No edge of S is present within C_ℓ.*
3. *For each edge $(u, v) \in S$ with $u \in C_\ell$, we have $v \in T'$.*
4. *If $T' \cap S^+ = \emptyset$, then $C_\ell \cap S^- = \emptyset$.*

Proof.   1. If C_ℓ does not contain any vertex from T, then the vertices of C_ℓ cannot reach any vertex of T in $G \setminus T'$. This means that C_ℓ is in the (reverse) shadow of T', which is a contradiction to the fact that T' is shadowless.

2. If C_ℓ contains an edge of S, then we will have an S-closed-walk in the strongly connected component C_ℓ, which is a contradiction, as T' is a solution for the instance (G, S, T, k) of DISJOINT SUBSET-DFVS COMPRESSION.

3. Consider an edge $(u, v) \in S$ such that $u \in C_\ell$ and $v \notin T'$. All outgoing edges from u must lie within C_ℓ, since C_ℓ is the last strongly connected component. In particular $v \in C_\ell$, which contradicts the second claim of the lemma.

4. Assume that $(u, v) \in S$ and $u \in C_\ell$ (which means $u \in C_\ell \cap S^-$). Since T' contains no vertex of S^+ we have $v \notin T'$ and by the third property we have $u \notin C_\ell$, a contradiction. $\qquad\square$

Lemma 5.2 suggests that we can start by guessing the (nonempty) subset $T_0 \subseteq T$ of vertices appearing in the last component C_ℓ. Given a set X of removed vertices, we say that edge $(u, v) \in S$ is *traversable* from T_0 in $G \setminus X$ if $u, v \notin X$ and vertex u (and hence v) is reachable from T_0 in $G \setminus X$.

22

If T' is a shadowless solution, then Lemma 5.2(2) implies that no edge of S is traversable from T_0 in $G \setminus T'$. There are two ways of making sure that an edge $(u, v) \in S$ is not traversable: (i) by making u unreachable from T_0, or (ii) by including v in T'. The situation is significantly simpler if every edge of S is handled the first way, that is, S^- is unreachable from T_0 in $G \setminus T'$. Then T' contains a $T_0 - S^-$ separator, and (as we shall see later) we may assume that T' contains an important $T_0 - S^-$ separator. Therefore, we can proceed by branching on choosing an important $T_0 - S^-$ separator of size at most k and including it into the solution.

The situation is much more complicated if some edges of S are handled the second way. Given a set X of vertices, we say that an edge $(u, v) \in S$ is *critical* (with respect to X) if $v \in X$ and u is reachable from T_0 in $G \setminus X$. Our main observation is that only a bounded number of vertices can be the head of a critical edge in a solution. Moreover, we can enumerate these vertices (more precisely, a bounded-size superset of these vertices) and therefore we can branch on including one of these vertices in the solution. We describe next how to enumerate these vertices.

Let us formalize the property of the vertices we are looking for:

Definition 5.3. (critical vertex) *For a fixed non-empty set $T_0 \subseteq V$, a vertex $v \in (V \setminus T_0) \cap S^+$ is called an ℓ-critical vertex, with respect to T_0, if there exists an edge $(u, v) \in S$ and a set $W \subseteq V \setminus T_0$ such that:*

- *$|W| \leq \ell$,*
- *edge (u, v) is critical with respect to W (that is, u is reachable from T_0 in $G \setminus W$ and $v \in W$),*
- *no edge of S is traversable from T_0 in $G \setminus W$.*

We say that v is witnessed by u, T_0 and W.

We need an upper bound on the number of critical vertices, furthermore our proof needs to be algorithmic, as we want to find the set of critical vertices, or at least a bounded-size superset of this. Roughly speaking, to test if v is a critical vertex, we need to check if there is a set T' that "cuts away" every edge of S from T_0 in a way that some vertex u with $(u, v) \in S$ is still reachable from T_0. One could argue that it is sufficient to look at important separators: if there is such a separator where u is reachable from T_0, then certainly there is an important separator where u is reachable from T_0. However, describing the requirement as "cutting away every edge of S from T_0" is imprecise: what we need is that no edge of S is traversable from T_0, which cannot be simply described by the separation of two sets of vertices. We fix this problem by moving to an auxiliary graph G' by duplicating vertices; whether or not an edge of S is traversable from T_0 translates to a simple reachability question in G'. However, due to technical issues that arise from this transformation, it is not obvious how to enumerate precisely the k-critical vertices. Instead, we construct a set F of bounded size that contains each k-critical vertex, and potentially some additional vertices. Thus if the solution has a critical edge, then we can branch on including a vertex of F into the solution.

Theorem 5.4. (bounding critical vertices) *Given a directed graph G, a subset S of its edges, and a fixed non-empty subset $T_0 \subseteq V(G)$, we can find in time $O^*(2^{O(k)})$ a set F_{T_0} of $2^{O(k)}$ vertices that is a superset of all k-critical vertices with respect to T_0.*

Proof. We create an auxiliary graph G', where the vertex set of G' consists of two copies for each vertex of V and two extra vertices s and t, i.e., $V(G') = \{v_{\text{in}}, v_{\text{out}} : v \in V\} \cup \{s, t\}$. The edges of G' are defined as follows (see also Fig. 6):

- For each edge $e = (u, v) \in E(G)$, we add the following edges to $E(G')$: if $e \in S$, then add to $E(G')$ an edge $(u_{\text{out}}, v_{\text{in}})$, otherwise add to $E(G')$ an edge $(u_{\text{out}}, v_{\text{out}})$.
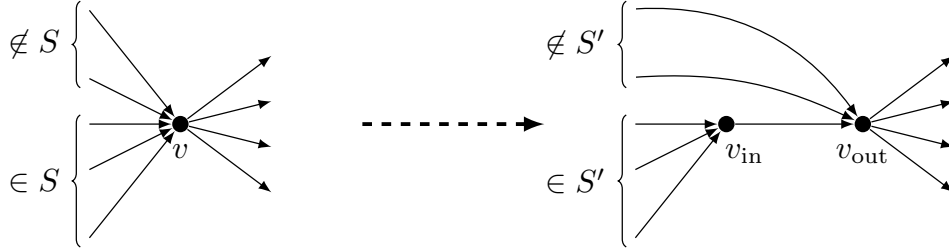
Figure 6: On the left there is a vertex v of G and on the right the corresponding vertices v_{in} and v_{out} of G'.

- For each vertex $v \in V$, we add to $E(G')$ an edge $(v_{\text{in}}, v_{\text{out}})$.
- For each vertex $v \in V$, we add an edge (v_{in}, t) to $E(G')$.
- For each vertex $v \in T_0$, we add an edge (s, v_{out}) to $E(G')$.

Let F'_{T_0} be the set of vertices of G' which belong to some important $s-t$ separator of size at most $2k$. By Lemma 3.8 the cardinality of F'_{T_0} is at most $2k \cdot 4^{2k}$. We define F_{T_0} as $\{v \in V : v_{\text{in}} \in F'_{T_0}\}$. Clearly, the claimed upper bound of $2^{O(k)}$ on $|F_{T_0}|$ follows, hence it remains to prove that each k-critical vertex belongs to F_{T_0}.

Let x be an arbitrary k-critical vertex witnessed by u, T_0 and W. Define $W' = \{v_{\text{in}}, v_{\text{out}} : v \in W\}$ and note that $|W'| \leq 2k$. The only out-neighbors of s are $\{v_{\text{out}} \mid v \in T_0\}$ while the only in-neighbors of t are $\{v_{\text{in}} \mid v \in V\}$. Hence the existence of an $s-t$ path in G' implies that there is in fact an edge $(a, b) \in S$ that is traversable from T_0 in $G \setminus W$ (at some point we have to go from an "out" vertex to an "in" vertex, and the only possible way to do this is via an edge from S). This is a contradiction to Definition 5.3. Therefore, no in-neighbor of t is reachable from s in $G' \setminus W'$, i.e., W' is an $s-t$ separator. Finally, a path from T_0 to u in $G \setminus W$ translates into a path from s to u_{out} in $G' \setminus W'$. Consider an important $s-t$ separator W'', i.e., $|W''| \leq |W'|$ and $R^+_{G' \setminus W'}(s) \subset R^+_{G' \setminus W''}(s)$. As u_{out} is reachable from s in $G' \setminus W'$ we infer that u_{out} is also reachable from s in $G' \setminus W''$. Consequently $x_{\text{in}} \in W''$, as otherwise there would be an $s-t$ path in $G' \setminus W''$. Hence x_{in} belongs to F'_{T_0}, which implies that x belongs to F_{T_0} and the theorem follows. $\qquad \square$

The following theorem characterizes a solution, so that we can find a vertex contained in it by inspecting a number of vertices in V bounded by a function of k. We apply Theorem 5.4 for each subset $T_0 \subseteq T$ and let $F = \bigcup_{T_0 \subseteq T} F_{T_0}$. Note that $|F| \leq 2^{|T|} \cdot 2^{O(k)} = 2^{O(|T|+k)}$, and we can generate F in time $2^{|T|} \cdot O^*(2^{O(k)}) = O^*(2^{O(|T|+k)})$

Theorem 5.5. (pushing) *Let $I = (G, S, T, k)$ be an instance of* DISJOINT SUBSET-DFVS COMPRESSION *having a shadowless solution and let F be a set generated by the algorithm of Theorem 5.4. Let G^+ be obtained from G by introducing a new vertex t and adding an edge (u, t) for every $u \in S^-$. Then there exists a solution $T' \subseteq V \setminus T$ for I such that either*

- *T' contains a vertex of $F \setminus T$, or*
- *T' contains an important $T_0 - (\{t\} \cup (T \setminus T_0))$ separator of G^+ for some non-empty $T_0 \subseteq T$.*

Proof. Let T' be any shadowless solution for I and let T_0 be the subset of T belonging to the last strongly connected component of $G \setminus T'$; by Property 1 of Lemma 5.2, T_0 is nonempty.

We consider two cases: either there is a $T_0 - S^-$ path in $G \setminus T'$ or not. First assume that there is a path from T_0 to a vertex $u \in S^-$ in $G \setminus T'$. Clearly, $u \in C_\ell$, since all vertices of T_0 belong to C_ℓ

and no edge from C_ℓ can go to previous strongly connected components. Consider any edge from S that has u as its starting point, say $(u,v) \in S$. By Property 3 of Lemma 5.2, we know that $v \in T'$. Observe that v is a k-critical vertex witnessed by u, T_0, and T', since $|T'| \leq k$, by definition of u, there is a path from T_0 to u in $G \setminus T'$; and by Property 3 of Lemma 5.2, no edge of S is traversable from T_0. Consequently, by the property of the set F, we know that $v \in T' \cap F \neq \emptyset$ and the theorem holds.

Now we assume that no vertex of S^- is reachable from T_0 in $G \setminus T'$. By the definition of T_0, the set T' is a $T_0 - (T \setminus T_0)$ separator in G, hence we infer that T' is a $T_0 - (\{t\} \cup (T \setminus T_0))$ separator in G^+. Let T^* be the subset of T' reachable from T_0 without going through any other vertices of T'. Then T^* is clearly a $T_0 - (\{t\} \cup (T \setminus T_0))$ separator in G^+. Let T^{**} be the minimal $T_0 - (\{t\} \cup (T \setminus T_0))$ separator contained in T^*. If T^{**} is an important $T_0 - (\{t\} \cup (T \setminus T_0))$ separator, then we are done, as T' itself contains T^{**}.

Otherwise, there exists an important $T_0 - (\{t\} \cup (T \setminus T_0))$ separator T^{***} that dominates T^{**}, i.e., $|T^{***}| \leq |T^{**}|$ and $R^+_{G^+ \setminus T^{**}}(T_0) \subset R^+_{G^+ \setminus T^{***}}(T_0)$. Now we claim that $T'' = (T' \setminus T^{**}) \cup T^{***}$ is a solution for the instance (G, S, T, k) of DISJOINT SUBSET-DFVS COMPRESSION. If we show this, then we are done, as $|T''| \leq |T'|$ and T'' contains the important $T_0 - (\{t\} \cup (T \setminus T_0))$ separator T^{***}.

Suppose T'' is a not a solution for the instance (G, S, T, k) of DISJOINT SUBSET-DFVS COMPRESSION. We have $|T''| \leq |T'| \leq k$ (as , $|T^{***}| \leq |T^{**}|$) and $T'' \cap T = \emptyset$ (as T^{***} is an important $T_0 - (\{t\} \cup (T \setminus T_0))$ separator of G^+, hence disjoint from T). Therefore, the only possible problem is that there is an S-closed-walk in $G \setminus T''$ passing through some vertex $v \in T^{**} \setminus T^{***}$; in particular, this implies that there is a $v - S^-$ walk in $G \setminus T''$. Since T^{**} is a minimal $T_0 - (\{t\} \cup (T \setminus T_0))$ separator and $R^+_{G^+ \setminus T^{**}}(T_0) \subset R^+_{G^+ \setminus T^{***}}(T_0)$, we have $(T^{**} \setminus T^{***}) \subseteq R^+_{G^+ \setminus T''}(T_0)$, implying $v \in R^+_{G^+ \setminus T''}(T_0)$. This gives a $T_0 - S^-$ walk via v in $G \setminus T''$, a contradiction as T'' contains an (important) $T_0 - (\{t\} \cup (T \setminus T_0))$ separator by construction. □

Theorem 5.5 tells us that there is always a minimum solution which either contains some critical vertex of F or an important $T_0 - (\{t\} \cup (T \setminus T_0))$ separator of G^+ where T_0 is a non-empty subset of T. In the former case, we branch into $|F|$ instances, in each of which we put one vertex of F to the solution, generating $2^{O(|T|+k)}$ instances with reduced budget. Next we can assume that the solution does not contain any vertex of F and we try all $2^{|T|} - 1$ choices for T_0. For each guess of T_0 we enumerate at most 4^k important $T_0 - (\{t\} \cup (T \setminus T_0))$ separators of size at most k in time $O^*(4^k)$ as given by Lemma 3.8. This gives the branching algorithm described in Algorithm 2.

6 Disjoint Subset-DFVS Compression: Summary of Algorithm

Lemma 4.6 and the BRANCH algorithm together combine to give a bounded search tree FPT algorithm for DISJOINT SUBSET-DFVS COMPRESSION described in Algorithm 3.

We then repeatedly perform Steps 1 and 2. Note that for every instance, one execution of steps 1 and 2 gives rise to $2^{O(k^2)} \log^2 n$ instances such that for each instance, either we know that the answer is NO or the budget k has decreased, because we have assumed that from each vertex of T one can reach the set S^-, and hence each important separator is non-empty. Therefore, considering a level as an execution of Step 1 followed by Step 2, the height of the search tree is at most k. Each time we branch into at most $2^{O(k^2)} \log^2 n$ directions (as $|T|$ is at most $k + 1$). Hence the total number of nodes in the search tree is $\left(2^{O(k^2)} \log^2 n \right)^k$.

Lemma 6.1. *For every n and $k \leq n$, we have $(\log n)^k \leq (2k \log k)^k + \frac{n}{2^k}$ (the logs are to base 2)*

Algorithm 2: BRANCH

Input: An instance $I = (G, S, T, k)$ of DISJOINT SUBSET-DFVS COMPRESSION.
Output: A new set of $2^{O(|T|+k)}$ instances of DISJOINT SUBSET-DFVS COMPRESSION where the budget k is reduced.

1: **for** every vertex $v \in F \setminus T$ found by Theorem 5.4 **do**
2:      Create a new instance $I_v = (G \setminus v, S, T, k-1)$ of DISJOINT SUBSET-DFVS COMPRESSION.
3: **for** every non-empty subset T_0 of T: **do**
4:      Use Lemma 3.8 to enumerate all the at most 4^k important $T_0 - (\{t\}^- \cup (T \setminus T_0))$ separators of size at most k in G^+.
5:      Let the important separators be $\mathcal{B} = \{B_1, B_2, \ldots, B_m\}$.
6:      **for** each $i \in [m]$ **do**
7:          Create a new instance $I_{T_0,i} = (G \setminus B_i, S, T, k - |B_i|)$ of DISJOINT SUBSET-DFVS COMPRESSION.

Algorithm 3: FPT Algorithm for Subset-DFVS

Step 1: For a given instance $I = (G, S, T, k)$, use Theorem 3.6 to obtain a set of instances $\{Z_1, Z_2, \ldots, Z_t\}$ where $t = 2^{O(k^2)} \log^2 n$ and Lemma 4.6 implies

- If I is a no-instance, then all the reduced instances $G_j = G/Z_j$ are no-instances for all $j \in [t]$
- If I is a yes-instance, then there is at least one $i \in [t]$ such that there is a solution T^* for I which is a shadowless solution for the reduced instance $G_i = G/Z_i$.

At this step we branch into $2^{O(k^2)} \log^2 n$ directions.

Step 2 : For each of the instances obtained from the above step, we run the BRANCH algorithm to obtain a set of $2^{O(k+|T|)}$ instances where in each case either the answer is NO, or the budget k is reduced. We solve these instances recursively and return YES if at least one of them returns YES.

Proof. If $\frac{\log n}{1+\log\log n} \geq k$, then $n \geq (2\log n)^k$. Otherwise we have $\frac{\log n}{1+\log\log n} < k$ and then $(4k\log k) \geq (2\log n)$ as follows: $2k\log k \geq \frac{2\log n \log k}{1+\log\log n}$. Now $\frac{2\log n \log k}{1+\log\log n} \geq \log n \Leftrightarrow 2\log k \geq 1 + \log\log n \Leftrightarrow k^2 \Leftrightarrow 2\log n$. But, $\frac{k^2}{2\log n} = \frac{\log n}{2(1+\log\log n)^2}$ which is greater than 1 for $n \geq 2^{2^7}$. $\square$

The total number of nodes in the search tree is $\left(2^{O(k^2)} \log^2 n\right)^k = \left(2^{O(k^2)}\right)^k (\log^2 n)^k = (2^{O(k^3)})(\log^2 n)^k \leq (2^{O(k^3)})\left((2k\log k)^k + \frac{n}{2^k}\right)^2 \leq 2^{O(k^3)} n^2$.

We then check the leaf nodes and see if there are any S-closed-walks left even after the budget k has become zero. If the graph in at least one of the leaf nodes is S-closed-walk free, then the given instance is a yes-instance. Otherwise it is a no-instance. This gives an $O^*(2^{O(k^3)})$ algorithm for DISJOINT SUBSET-DFVS COMPRESSION. By Lemma 2.3, we have an $O^*(2^{O(k^3)})$ algorithm for the SUBSET-DFVS problem.

7 Conclusion and Open Problems

In this paper we gave the first fixed-parameter algorithm for DIRECTED SUBSET FEEDBACK VERTEX SET parameterized by the size of the solution. Our algorithm used various tools from the FPT world such as iterative compression, bounded-depth search trees, random sampling of important separators, etc. We also gave a general family of problems for which we can do random sampling of important separators and obtain a set which is disjoint from a minimum solution and covers its shadow. We believe this general approach will be useful for deciding the fixed-parameter tractability status of other problems in directed graphs, where we do not know that many techniques unlike undirected graphs.

The next natural question is whether SUBSET-DFVS has a polynomial kernel or can we rule out such a possibility under some standard assumptions? The recent developments [12, 30, 29] in the field of kernelization may be useful in answering this question. In the field of exact exponential algorithms, Razgon [41] gave an $O^*(1.9977^n)$ algorithm for DFVS which was used by Chitnis et al. [10] to give an $O^*(1.9993^n)$ algorithm for the more general SUBSET-DFVS problem. It would be interesting to improve either of this algorithms.

Acknowledgements

References

[1] Bafna, V., Berman, P., Fujito, T.: A 2-Approximation Algorithm for the Undirected Feedback Vertex Set Problem. SIAM J. Discrete Math. 12(3), 289–297 (1999)

[2] Becker, A., Bar-Yehuda, R., Geiger, D.: Randomized Algorithms for the Loop Cutset Problem. J. Artif. Intell. Res. (JAIR) 12, 219–234 (2000)

[3] Bodlaender, H.L.: On Disjoint Cycles. In: WG. pp. 230–238 (1991)

[4] Bonsma, P., Lokshtanov, D.: Feedback Vertex Set in Mixed Graphs. In: WADS. pp. 122–133 (2011)

[5] Cao, Y., Chen, J., Liu, Y.: On Feedback Vertex Set: New Measure and New Structures. In: SWAT. pp. 93–104 (2010)

[6] Chen, J., Fomin, F.V., Liu, Y., Lu, S., Villanger, Y.: Improved algorithms for feedback vertex set problems. J. Comput. Syst. Sci. 74(7), 1188–1198 (2008)

[7] Chen, J., Liu, Y., Lu, S.: An Improved Parameterized Algorithm for the Minimum Node Multiway Cut Problem. Algorithmica 55(1), 1–13 (2009)

[8] Chen, J., Liu, Y., Lu, S., O'Sullivan, B., Razgon, I.: A fixed-parameter algorithm for the directed feedback vertex set problem. J. ACM 55(5) (2008)

[9] Chitnis, R.H., Cygan, M., Hajiaghayi, M.T., Marx, D.: Directed subset feedback vertex set is fixed-parameter tractable. In: ICALP (1). pp. 230–241 (2012)

[10] Chitnis, R.H., Fomin, F.V., Lokshtanov, D., Misra, P., Ramanujan, M.S., Saurabh, S.: Faster exact algorithms for some terminal set problems. In: IPEC. pp. 150–162 (2013)

[11] Chitnis, R.H., Hajiaghayi, M., Marx, D.: Fixed-Parameter Tractability of Directed Multiway Cut Parameterized by the Size of the Cutset. SIAM J. Comput. 42(4), 1674–1696 (2013)

[12] Cygan, M., Kratsch, S., Pilipczuk, M., Pilipczuk, M., Wahlström, M.: Clique Cover and Graph Separation: New Incompressibility Results. TOCT 6(2), 6 (2014)

[13] Cygan, M., Nederlof, J., Pilipczuk, M., Pilipczuk, M., van Rooij, J.M.M., Wojtaszczyk, J.O.: Solving Connectivity Problems Parameterized by Treewidth in Single Exponential Time. In: FOCS. pp. 150–159 (2011)

[14] Cygan, M., Pilipczuk, M., Pilipczuk, M., Wojtaszczyk, J.O.: On multiway cut parameterized above lower bounds. TOCT 5(1), 3 (2013)

[15] Cygan, M., Pilipczuk, M., Pilipczuk, M., Wojtaszczyk, J.O.: Subset feedback vertex set is fixed-parameter tractable. SIAM J. Discrete Math. 27(1), 290–309 (2013)

[16] Dehne, F., Fellows, M.R., Langston, M.A., Rosamond, F.A., Stevens, K.: An $O(2^{O(k)}n^3)$ FPT Algorithm for the Undirected Feedback Vertex Set Problem. Theory Comput. Syst. 41(3), 479–492 (2007)

[17] Downey, R.G., Fellows, M.R.: Fixed-Parameter Tractability and Completeness I: Basic Results. SIAM J. Comput. 24(4), 873–921 (1995)

[18] Downey, R.G., Fellows, M.R.: Parameterized Complexity. Springer-Verlag (1999), 530 pp.

[19] Even, G., Naor, J., Schieber, B., Sudan, M.: Approximating Minimum Feedback Sets and Multi-Cuts in Directed Graphs. In: IPCO. pp. 14–28 (1995)

[20] Even, G., Naor, J., Zosin, L.: An 8-Approximation Algorithm for the Subset Feedback Vertex Set Problem. SIAM J. Comput. 30(4), 1231–1252 (2000)

[21] Flum, J., Grohe, M.: Parameterized Complexity Theory. Springer-Verlag (2006), 493 pp.

[22] Fomin, F.V., Gaspers, S., Kratsch, D., Liedloff, M., Saurabh, S.: Iterative compression and exact algorithms. Theor. Comput. Sci. 411(7-9), 1045–1053 (2010)

[23] Guo, J., Gramm, J., Hüffner, F., Niedermeier, R., Wernicke, S.: Compression-based fixed-parameter algorithms for feedback vertex set and edge bipartization. J. Comput. Syst. Sci. 72(8), 1386–1396 (2006)

[24] Hüffner, F., Komusiewicz, C., Moser, H., Niedermeier, R.: Fixed-Parameter Algorithms for Cluster Vertex Deletion. In: LATIN. pp. 711–722 (2008)

[25] Kakimura, N., Kawarabayashi, K., Kobayashi, Y.: Erdös-pósa property and its algorithmic applications: parity constraints, subset feedback set, and subset packing. In: SODA. pp. 1726–1736 (2012)

[26] Kanj, I.A., Pelsmajer, M.J., Schaefer, M.: Parameterized Algorithms for Feedback Vertex Set. In: IWPEC. pp. 235–247 (2004)

[27] Karp, R.M.: Reducibility Among Combinatorial Problems. In: Complexity of Computer Computations. pp. 85–103 (1972)

[28] Kratsch, S., Pilipczuk, M., Pilipczuk, M., Wahlström, M.: Fixed-parameter tractability of multicut in directed acyclic graphs. In: ICALP (1). pp. 581–593 (2012)

[29] Kratsch, S., Wahlström, M.: Compression via matroids: a randomized polynomial kernel for odd cycle transversal. In: SODA. pp. 94–103 (2012)

[30] Kratsch, S., Wahlström, M.: Representative Sets and Irrelevant Vertices: New Tools for Kernelization. In: FOCS. pp. 450–459 (2012)

[31] Lokshtanov, D., Marx, D.: Clustering with local restrictions. Inf. Comput. 222, 278–292 (2013)

[32] Lokshtanov, D., Narayanaswamy, N.S., Raman, V., Ramanujan, M.S., Saurabh, S.: Faster parameterized algorithms using linear programming. ACM Transactions on Algorithms 11(2), 15 (2014)

[33] Lokshtanov, D., Ramanujan, M.S.: Parameterized Tractability of Multiway Cut with Parity Constraints. In: ICALP (1). pp. 750–761 (2012)

[34] Marx, D.: Parameterized graph separation problems. Theor. Comput. Sci. 351(3), 394–406 (2006)

[35] Marx, D., Razgon, I.: Fixed-parameter tractability of multicut parameterized by the size of the cutset. SIAM J. Comput. 43(2), 355–388 (2014)

[36] Mehlhorn, K.: Data Structures and Algorithms 2: Graph Algorithms and NP-completeness. Springer (1984)

[37] Naor, M., Schulman, L.J., Srinivasan, A.: Splitters and Near-Optimal Derandomization. In: FOCS. pp. 182–191 (1995)

[38] Niedermeier, R.: Invitation to Fixed-Parameter Algorithms. Oxford University Press (2006)

[39] Raman, V., Saurabh, S., Subramanian, C.R.: Faster Fixed Parameter Tractable Algorithms for Undirected Feedback Vertex Set. In: ISAAC. pp. 241–248 (2002)

[40] Raman, V., Saurabh, S., Subramanian, C.R.: Faster fixed parameter tractable algorithms for finding feedback vertex sets. ACM Transactions on Algorithms 2(3), 403–415 (2006)

[41] Razgon, I.: Computing Minimum Directed Feedback Vertex Set in $O(1.9977^n)$. In: ICTCS. pp. 70–81 (2007)

[42] Razgon, I., O'Sullivan, B.: Almost 2-SAT is fixed-parameter tractable. J. Comput. Syst. Sci. 75(8), 435–450 (2009)

[43] Reed, B.A., Smith, K., Vetta, A.: Finding odd cycle transversals. Oper. Res. Lett. 32(4), 299–301 (2004)

[44] Seymour, P.D.: Packing Directed Circuits Fractionally. Combinatorica 15(2), 281–288 (1995)