Subset Feedback Vertex Set on Graphs of Bounded Independent Set Size

Charis Papadopoulos† Spyridon Tzimas‡

Abstract

The (Weighted) Subset Feedback Vertex Set problem is a generalization of the classical Feedback Vertex Set problem and asks for a vertex set of minimum (weight) size that intersects all cycles containing a vertex of a prespecified set of vertices. Although Subset Feedback Vertex Set and Feedback Vertex Set exhibit different computational complexity on split graphs, no similar characterization is known on other classes of graphs. Towards the understanding of the complexity difference between the two problems, it is natural to study the importance of structural graph parameters. Here we consider graphs of bounded independent set number for which it is known that Weighted Feedback Vertex Set can be solved in polynomial time. We provide a dichotomy result with respect to the size $\alpha$ of a maximum independent set. In particular we show that Weighted Subset Feedback Vertex Set can be solved in polynomial time for graphs with $\alpha \leq 3$, whereas we prove that the problem remains NP-hard for graphs with $\alpha \geq 4$. Moreover, we show that the (unweighted) Subset Feedback Vertex Set problem can be solved in polynomial time on graphs of bounded independent set number by giving an algorithm with running time $n^{O(\alpha)}$.

To complement our results, we demonstrate how our ideas can be extended to other terminal set problems on graphs of bounded independent set size. Node Multiway Cut is a terminal set problem that asks for a vertex set of minimum size that intersects all paths connecting any two terminals. Based on our findings for Subset Feedback Vertex Set, we settle the complexity of Node Multiway Cut as well as its variants where nodes are weighted and/or the terminals are deletable, for every value of the given independent set number.

1 Introduction

Given a (vertex-weighted) graph $G = (V,E)$ and a set $S \subseteq V$, the (Weighted) Subset Feedback Vertex Set problem asks for a vertex set of minimum (weight) size that intersects all cycles containing a vertex of $S$. Subset Feedback Vertex Set was introduced in the context of approximation algorithms by Even et al. who obtained an 8-approximation algorithm for its weighted version [14]. Cygan et al. [12] and Kawarabayashi and Kobayashi [25] independently showed that Subset Feedback Vertex Set is fixed-parameter tractable (FPT) parameterized by the solution size, while Hols and Kratsch provided a randomized polynomial kernel for the problem [21]. There has been a considerable

---

*‡A preliminary version of this paper appeared as an extended abstract in the proceedings of IPEC 2018.
The research work done by C. Papadopoulos was supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.) under the “First Call for H.F.R.I. Research Projects to support Faculty members and Researchers and the procurement of high-cost research equipment grant” (Project Number: 431). The research work done by S. Tzimas has been financially supported by the General Secretariat for Research and Technology (G.S.R.T.) and the Hellenic Foundation for Research and Innovation (H.F.R.I.) (Scholarship Code: 1486).

†Department of Mathematics, University of Ioannina, Greece. E-mail: charis@uoi.gr

‡Department of Mathematics, University of Ioannina, Greece. E-mail: royzimas@hotmail.com
amount of work to obtain faster, still exponential-time, algorithms even when restricted to particular graph classes [6, 17, 16, 20, 33]. As a generalization of the classical Feedback Vertex Set for which $S = V$, the problem remains NP-hard on bipartite graphs [37] and planar graphs [18]. On the positive side, Weighted Subset Feedback Vertex Set can be solved in polynomial time on interval graphs, permutation graphs, and cobipartite graphs [32], the latter being a subclass of graphs of independent set size at most two. However a notable difference between the two problems regarding their complexity status is the class of split graphs: Feedback Vertex Set is known to be polynomial-time solvable on split graphs [8, 35], whereas Subset Feedback Vertex Set remains NP-hard on split graphs [17].

In order to obtain further (in)tractability results for Subset Feedback Vertex Set, it is reasonable to consider structural parameters of graphs that may lend themselves to provide a unified approach. In terms of parameterized complexity Feedback Vertex Set is known to be FPT, when parameterized by tree-width [10] and clique-width [2] which implies that Feedback Vertex Set can be solved in polynomial time on graphs of bounded such parameters. Although Feedback Vertex Set is W[1]-hard parameterized by the size of a maximum independent set\(^1\), it can be solved in polynomial time on graphs of bounded maximum induced matching (i.e., Feedback Vertex Set belongs in XP parameterized by the size of a maximum induced matching) [24]. Only very recently, Jaffke et al. proposed an algorithm that solves Weighted Feedback Vertex Set in time $n^{O(w)}$ where $w$ is the maximum induced matching width of the given graph [23]. Independently from the work of [23], Bergougnoux and Kanté proposed the same result through the notion of neighbor equivalence [1]. Despite their relevant name, graphs of bounded maximum induced matching (or graphs of bounded independent set number) are not related to graphs of bounded maximum induced matching width as indicated in [36].

The approach of [23] provides a powerful mechanism, as it unifies polynomial-time algorithms for Weighted Feedback Vertex Set on several graph classes such as interval graphs, permutation graphs, circular-arc graphs, and Dilworth-$k$ graphs for fixed $k$, among others. Such a mechanism raises the question of whether the algorithm given in [23] can be extended to the more general setting of Weighted Subset Feedback Vertex Set. However the proposed algorithm is based on the crucial fact that the forest formed by deleting the nodes of a solution has bounded number of internal nodes which is not necessarily true for the $S$-forest of Weighted Subset Feedback Vertex Set. Thus it seems difficult to control the size of the solution whenever $S \subset V$. As this observation does not rule out any positive answer, here we develop the first step towards such an approach by considering graphs of bounded independent set number which form candidate relevant graphs. Although Weighted Feedback Vertex Set can be solved in time $n^{O(p)}$ on graphs of maximum induced matching at most $p$ [24], Subset Feedback Vertex Set is already NP-complete on graphs of maximum induced matching equal to one (i.e., split graphs) [17]. Moreover, by standard algorithmic techniques, Weighted Subset Feedback Vertex Set is FPT when parameterized by the tree-width of the input graph.

In this work we show that the complexity behaviour of the weighted version of the problem is completely different from the behaviour of the unweighted variant on graphs of independent set number $\alpha$.

- We show that Weighted Subset Feedback Vertex Set can be solved in polynomial time on graphs with $\alpha \leq 3$.

Notice that graphs with $\alpha \leq 3$ properly contain the complements of triangle-free graphs and recall that for triangle-free graphs Feedback Vertex Set remains NP-hard [37]. We solve

\(^1\)In Section 4 we give a new and simpler reduction from the Multicolored Independent Set problem.
**Weighted Subset Feedback Vertex Set** on such graphs, by exploiting a structural characterization of the solution with respect to the vertices that are close to S.

- We further provide a dichotomy result showing that **Weighted Subset Feedback Vertex Set** remains NP-complete on graphs with \( \alpha \geq 4 \).

Thus we enlarge our knowledge on the complexity difference of the two problems with respect to a structural graph parameter.

- In order to complement our results, we show that **Subset Feedback Vertex Set** can be solved in \( n^{O(\alpha)} \) time for any value of \( \alpha \). We further show that the running time of our algorithm achieves a tight lower bound, under the Exponential Time Hypothesis. The latter result is accomplished through a W[1]-hardness reduction that preserves a linear blowup of the bigger parameter of clique cover number, combined with the conditional lower bound given in [4].

Our findings concerning **Subset Feedback Vertex Set** are summarized in Table 1.

Moreover, we demonstrate how our ideas can be extended to other terminal set problems on graphs of bounded independent set size. In the (unweighted) **Node Multiway Cut** problem, we are given a graph \( G = (V, E) \), a terminal set \( T \subseteq V \), and a nonnegative integer \( k \) and the goal is to find a set \( X \subseteq V \setminus T \) of size at most \( k \) such that any path between two different terminals intersects \( X \). **Node Multiway Cut** is known to be in FPT parameterized by the solution size [5, 29] and even above guaranteed value [11]. For further results on variants of **Node Multiway Cut** we refer to [3, 19, 27]. We completely characterize the complexity of **Node Multiway Cut** with respect to the size of the maximum independent set.

- In particular, we show that for \( \alpha \leq 2 \) **Node Multiway Cut** can be solved in polynomial time, whereas for \( \alpha \geq 3 \) it remains NP-complete by adapting the reduction for **Weighted Subset Feedback Vertex Set** for \( \alpha \geq 4 \).

We further consider a relaxed variation of **Node Multiway Cut** in which we are allowed to remove terminal vertices, called **Node Multiway Cut with Deletable Terminals** (also known as **Unrestricted Node Multiway Cut**).

- We show that the (unweighted) **Node Multiway Cut with Deletable Terminals** problem can be solved in polynomial time on graphs of bounded independent set number, using an idea similar to the polynomial-time algorithm for the **Subset Feedback Vertex Set** problem.

- Regarding its node-weighted variation, we provide a complexity dichotomy result showing that **Weighted Node Multiway Cut with Deletable Terminals** can be solved in polynomial time on graphs with \( \alpha \leq 2 \), whereas it becomes NP-complete on graphs with \( \alpha \geq 3 \).

We note that the polynomial-time algorithm for the weighted variation is obtained by invoking our algorithm for **Weighted Subset Feedback Vertex Set** on graphs with \( \alpha \leq 3 \).

## 2 Preliminaries

All graphs considered here are simple and undirected. A graph is denoted by \( G = (V, E) \) with vertex set \( V \) and edge set \( E \). We use the convention that \( n = |V| \) and \( m = |E| \). For a set of vertices \( X \subseteq V \), the subgraph of \( G \) induced by \( X \) is \( G[X] = (X, \{uw \in E : u, v \in X\}) \).
Max. Induced Matching (\(np\text{-complete} [17]\))

Vertex set \(X\) if there is no edge between any pair of these vertices, and it is an independent set \(w\) that is a positive integer number. We denote by \(\text{graph induced by the vertices of} \ V\) if there is a path between any pair of vertices. A connected graph is possible edges are present between pairs of these vertices. We say that a graph is bounded maximum induced matching. Note that every graph of independent set number \(\alpha\) has maximum induced matching of size at most \(\alpha\), while the converse is not necessarily true (e.g., star graphs).

\(X\}). The neighborhood of a vertex \(x\) of \(G\) is \(N_G(x) = \{v \in V : xv \in E\}\). For \(X \subseteq V\), \(N_G(X) = \bigcup_{v \in X} N_G(v) \setminus X\) and \(N_G[X] = N_G(X) \cup X\). Moreover, we denote by \(G - X\) the graph induced by the vertices of \(V \setminus X\).

A weighted graph \(G = (V,E)\) is a graph, where each vertex \(v \in V\) is assigned a weight that is a positive integer number. We denote by \(w(v)\) the weight of each vertex \(v \in V\). For a vertex set \(X \subseteq V\), the weight of \(X\), denoted by \(w(X)\), is \(\sum_{v \in X} w(v)\). A set of vertices is an independent set if there is no edge between any pair of these vertices, and it is a clique if all possible edges are present between pairs of these vertices. We say that a graph is connected if there is a path between any pair of vertices. A connected component of \(G\) is a maximal connected subgraph of \(G\). A forest is a graph that contains no cycles and a tree is a forest that is connected. For a positive integer \(k\), a \(k\)-partite graph is a graph whose vertices can be partitioned into \(k\) independent sets. When \(k = 2\) or \(k = 3\), the particular \(k\)-partite graphs are called bipartite and tripartite graphs, respectively.

Given a graph \(G\), the independent set number, denoted by \(\alpha(G)\), is the size of a maximum independent set in \(G\). The clique cover number of \(G\), denoted by \(\kappa(G)\), is the smallest number of cliques needed to partition \(V\) into \(S_1,\ldots,S_k\) such that \(G[S_i]\) is a clique. A vertex cover is a set of vertices such that every edge of \(G\) is incident to at least one vertex of the set. A matching is a set of edges having no common endpoint. An induced matching is a matching \(M\) of \(p\) edges such that \(G[V(M)]\) contains only the edges of \(M\). The maximum induced matching number, denoted by \(\mu(G)\), is the largest number of edges in any induced matching of \(G\). It is not difficult to see that for any graph \(G\), \(\kappa(G) \geq \alpha(G) \geq \mu(G)\) holds.

Here we consider the following problem.

\(- (\text{Weighted}) \text{ Subset Feedback Vertex Set - SFVS} -\)

**Input:** A (vertex-weighted) graph \(G\), a set \(S \subseteq V\), and a nonnegative integer \(k\).

**Task:** Decide whether there is a set \(X \subseteq V\) with \(|X| \leq k\) \((w(X) \leq k)\) such that no cycle in \(G - X\) contains a vertex of \(S\).

As remarked, we distinguish between the weighted and the unweighted version of the problem. In the unweighted version of the problem note that all weights are equal and positive. The classical Feedback Vertex Set (FVS) problem is a special case of Subset Feedback Vertex Set with \(S = V\). A vertex of \(S\) is simply called \(S\)-vertex. An induced cycle of \(G\) is called \(S\)-cycle if an \(S\)-vertex is contained in the cycle. An induced subgraph \(F\) of \(G\) is called \(S\)-forest if there is no \(S\)-cycle in \(F\). It is not difficult to see that the problem of computing a minimum weighted subset feedback vertex set is equivalent to the problem of computing a
maximum weighted $S$-forest.

Let us give a couple of observations on the nature of \textsc{Subset Feedback Vertex Set} on graphs of bounded independent set size. Firstly note that the bound on the size of an independent set is a hereditary property; for every induced subgraph $H$ of $G$, we have $\alpha(H) \leq \alpha(G)$. Moreover for any clique $C$ of $G$, any $S$-forest of $G$ contains at most two vertices of $S \cap C$.

Observation 2.1. Let $G$ be a graph with $\alpha = \alpha(G)$ and let $S \subseteq V$.

(1) For any set $X$ of at least $2\alpha + 1$ vertices, there is a cycle in $G[X]$.

(2) Any $S$-forest of $G$ has at most $2\alpha$ vertices from $S$.

Proof. Let $X$ be a set of at least $2\alpha + 1$ vertices. Assume that $G[X]$ is a forest. As an induced subgraph of $G$, any independent set of $G[X]$ has size at most $\alpha$. Since $G[X]$ is acyclic, there is a proper 2-coloring $A, B$ of the vertices of $G[X]$ such that $|A| \geq |B|$. By the fact that $|A| \leq \alpha$, we conclude that $|A| + |B| \leq 2\alpha$, leading to a contradiction that $|X| \geq 2\alpha + 1$. Thus $G[X]$ contains a cycle.

For the second statement, let $F = (V_F, E_F)$ be an $S$-forest of $G$. By the first statement, if $S \cap V_F$ has at least $2\alpha + 1$ vertices then there is a cycle in $F[S \cap V_F]$, which implies an $S$-cycle in $F$. Thus $|S \cap V_F| \leq 2\alpha$. \hfill \QED

We note that Observation 2.1 allows us to construct by brute force all possible subsets of $S$ belonging to any $S$-forest in $n^{O(\alpha)}$ time.

We conclude this section by giving some definitions from the parameterized complexity theory; for further details we refer to [9, 13]. Parameterized complexity is a two dimensional framework for studying the computational complexity of a problem. One dimension is the input size $n$ and the other is a parameter $k$ associated with the input. The complexity class XP is composed by all parameterized problems with input size $n$ and parameter $k$ that can be solved in time $n^{f(k)}$ for some computable function $f$. A parameterized problem is \textit{para-NP-hard} if it is already NP-hard for a constant value of $k$. It is known that a para-NP-hard problem cannot belong to XP, unless P=NP. A problem with input size $n$ and parameter $k$ is \textit{fixed parameter tractable} (FPT), if it can be solved in time $f(k) \cdot n^{O(1)}$ for some computable function $f$. Respectively, the complexity class FPT is composed by all fixed parameter tractable problems. Parameterized complexity also provides tools to refute the FPT algorithms under plausible complexity-theoretic assumptions. The main assumptions is the conjecture that FPT $\neq W[1]$ for the parameterized complexity class W[1]. The basic way to show that it is unlikely that a parameterized problem admit an FPT algorithm is to show that it is W[1]-hard using a \textit{parameterized reduction} from a known W[1]-hard problem.

3 Weighted SFVS on Graphs of Bounded Independent Set Size

Here we consider the Weighted \textsc{Subset Feedback Vertex Set} and we show a dichotomy result with respect to the size of the maximum independent set. We first provide a polynomial-time algorithm on graphs of independent set size at most three and then we show that Weighted \textsc{Subset Feedback Vertex Set} is NP-complete on graphs of independent set size at least four.

Let $(G, S, k)$ be an instance of Weighted \textsc{Subset Feedback Vertex Set} where $G$ is a graph of independent set size $\alpha = \alpha(G)$. In the forthcoming arguments, instead of directly computing a solution for Weighted \textsc{Subset Feedback Vertex Set}, we consider the equivalent problem of computing an $S$-forest of $G$ having weight at least $w(V) - k$. 

5
Let $H = (V_H, E_H)$ be an induced subgraph of $G$. Let $S_0 = S \cap V_H$ and let $S_1 = N_H(S_0)$. Furthermore, we denote by $S_{\leq 1}$ the set $S_0 \cup S_1$. We partition the graph $H$ into two induced subgraphs $H_{\leq 1}$ and $H_{> 1}$ as follows:

- $H_{\leq 1}$ is the subgraph $H[S_{\leq 1}]$ of $H$ that is induced by the vertices that are at distance at most one from the vertices of $S_0$.
- $H_{> 1}$ is the subgraph $H - S_{\leq 1}$ of $H$ that is induced by the vertices that are at distance at least two from the vertices of $S_0$.

Such a partition is called the $S$-distance partition of $H$, denoted by $(H_{\leq 1}, H_{> 1})$. The set of edges of $H$ having one endpoint in $H_{\leq 1}$ and the other in $H_{> 1}$ is called the cut of the partition $(H_{\leq 1}, H_{> 1})$. Notice that a vertex of $H_{\leq 1}$ that is adjacent to a vertex of $H_{> 1}$ belongs to $S_1$.

Let $(C_1, \ldots, C_d)$ be an ordering of the partition of the vertices of $H_{> 1}$ such that each $C_i$, $1 \leq i \leq d$, induces a connected component in $H_{> 1}$. Because $H_{> 1}$ is an induced subgraph of $G$, it is clear that $d \leq \alpha$. Let $(A_1, \ldots, A_d)$ be a tuple of $d$ subsets of $S_1$, i.e., each $A_i \subseteq S_1$ holds. Observe that $(A_1, \ldots, A_d)$ neither partitions nor covers the set $S_1$. We say that the cut satisfies the tuple $(A_1, \ldots, A_d)$ if for any vertex $v \in C_i$, we have $(N(v) \cap S_{\leq 1}) \subseteq A_i$. Recall that an $S$-forest is an induced subgraph of $G$. Thus, an $S$-forest $F$ of $G$ admits an $S$-distance partition $(F_{\leq 1}, F_{> 1})$. The notion of an $S$-distance partition of $F$ with the corresponding cut is illustrated in Figure 1.

We now utilize the $S$-distance partition of $H$ in order to construct an algorithm that solves **Weighted Subset Feedback Vertex Set** on graphs of independent set size $\alpha$ and subsequently show that this algorithm is efficient for $\alpha \leq 3$. Our general approach relies on the following facts:

- By Observation 2.1 (2), we try all subsets $S'$ of $S$ with at most $2\alpha$ vertices and keep those sets that induce a forest. This step is used in constructing the vertices of $S$ within the graph $H_{\leq 1}$. In particular, for each such set $S'$, we construct all $H_{\leq 1}$ such that $S_0 = S'$. We will show that the number of such subsets produced is bounded by $n^{O(\alpha)}$.
- For each of the potential subgraphs $H_{\leq 1}$ constructed in the previous step, and for each $d \leq \alpha$, we determine all possible tuples $(A_1, \ldots, A_d)$ with $A_i \subseteq S_1$ having the following property: every induced subgraph of $G$ whose $S$-distance partition’s first part is $H_{\leq 1}$ and its cut satisfies the tuple $(A_1, \ldots, A_d)$ is indeed an $S$-forest $F$ of $G$. We show that considering only these tuples is sufficient in Lemma 3.1.
• Up to that point, we can show that all steps can be executed in time $n^{O(\alpha)}$. However for the next and final step we can only achieve polynomial running time if we restrict ourselves to $\alpha \leq 3$ due to the number of connected components of $F_{>1}$. For each tuple computed in the previous step, we find connected components $C_1, \ldots, C_d$ of maximum cumulative weight such that the cut of $(F_{\leq 1}, G[C_1 \cup \cdots \cup C_d])$ satisfies the tuple. For doing so, we take advantage of the small number of connected components ($d \leq 3$) and an efficient way of computing a vertex-cut between such components.

We begin by showing that the $S$-distance partition of $H$ provides a useful tool towards computing a maximum $S$-forest. Given a set of vertices $X \subseteq N[S]$ and $d$ subsets $A_i$ of $X \setminus S$, we construct the graph $aux_{A_1, \ldots, A_d}(X)$ that is obtained from $G[X]$ by adding $d$ vertices $w_1, \ldots, w_d$ such that every vertex $w_i$ is adjacent to all the vertices of $A_i$. In what follows, we always assume that $G$ is a graph having independent set size $\alpha$.

**Lemma 3.1.** Let $F$ be an $S$-forest of $G$ with $S$-distance partition $(F_{\leq 1}, F_{>1})$ such that $S_0 \neq \emptyset$. Then for some $\alpha \leq d$, there is a tuple $(A_1, \ldots, A_d)$ with $A_i \subseteq S_1$ such that

(i) the cut of $(F_{\leq 1}, F_{>1})$ satisfies $(A_1, \ldots, A_d)$ and

(ii) every induced subgraph $H$ of $G$ with $S$-distance partition $(H[S_{\leq 1}], H[S_{>1}])$ that satisfies $(A_1, \ldots, A_d)$ is an $S$-forest.

**Proof.** Let $(C_1, \ldots, C_d)$ be an ordering of the partition of the vertices of $F_{>1}$ such that every $C_i$ induces a connected component in $F_{>1}$. We define a tuple $(A_1, \ldots, A_d)$ in which every $A_i = N(C_i) \cap S_{\leq 1}$, for $1 \leq i \leq d$. Clearly $A_i \subseteq S_1$ since every vertex $F_{>1}$ is at distance at least two from $S_0$. Thus, by construction, the cut of $(F_{\leq 1}, F_{>1})$ satisfies the tuple $(A_1, \ldots, A_d)$.

For the next claim, we first show that $G := aux_{A_1, \ldots, A_d}(S_{\leq 1})$ is an $S$-forest. Assume for contradiction that there is an $S$-cycle $\hat{C}$ in $G$. Since $F_{\leq 1}$ does not contain any $S$-cycle, $\hat{C}$ contains a vertex $w_i$ and at least two vertices $u_i, v_i$ from $A_i$, for some $1 \leq i \leq d$. By the fact that $A_i = N(C_i) \cap S_{\leq 1}$, there are vertices $x_i$ and $y_i$, not necessarily distinct, in $C_i$ that are adjacent to $u_i$ and $v_i$, respectively. Since $C_i$ induces a connected component of $F_{>1}$, there is a path between $x_i$ and $y_i$, that lies entirely in $F_{>1}[C_i]$. This means that we can replace the vertex $w_i$ of $\hat{C}$ by a path between $x_i$ and $y_i$ for every $i$, to obtain an $S$-cycle in $F$, leading to a contradiction. Thus, $G$ is an $S$-forest.

Let $H$ be an induced subgraph of $G$ with $S$-distance partition $(H[S_{\leq 1}], H[S_{>1}])$ that satisfies $(A_1, \ldots, A_d)$. Observe that $H[S_{\leq 1}] = F_{\leq 1}$ as they are induced subgraphs of the same vertex set of $G$. Thus $H[S_{\leq 1}]$ does not contain any $S$-cycle, because $F$ is an $S$-forest. Since the cut of $(H[S_{\leq 1}], H[S_{>1}])$ satisfies $(A_1, \ldots, A_d)$, there is a partition $(T_1, \ldots, T_d)$ of the vertices of $H[S_{>1}]$ such that $T_i$ is a connected component of $H[S_{\leq 1}]$ and $N(T_i) \subseteq A_i$, for $1 \leq i \leq d$. We show that $H$ is indeed an $S$-forest. For contradiction, assume an $S$-cycle $C$ in $H$. There are no $S$-cycles in $H[S_{\leq 1}]$ which implies that $C \cap T_i \neq \emptyset$, for some $1 \leq i \leq d$. For every such set, we replace the part $C \cap T_i$ by a vertex $w'_i$. Denote by $H'$ the resulting graph. Notice that $H'[C]$ is a subgraph of $\hat{G}[C]$ because $N(H'[w'_i]) \subseteq N_{\hat{G}}(w_i)$. This, however, implies an $S$-cycle in $\hat{G}$, which gives the desired contradiction. Therefore, $H$ is an $S$-forest.

Notice that $G - S$ is trivially an $S$-forest of $G$. Moreover, $G - S$ is maximal among all $S$-forests of $G$ such that $S_0 = \emptyset$. In what follows, we assume that $S_0 \neq \emptyset$ and show how to bound the vertex set $S_{\leq 1}$ of $F_{\leq 1}$.

**Lemma 3.2.** Let $F$ be an $S$-forest of $G$ such that $S_0 \neq \emptyset$.

1. If $|S_0| \leq 2\alpha - 2$ then $|S_{\leq 1}| \leq 4\alpha - 2$.

2. If $|S_0| \geq 2\alpha - 1$ then $|S_{\leq 1}| \leq 2\alpha$. 

7
that the induced cycle of $F$ implies that there is a cycle in $F \setminus W$ of $S$.

Case 2. Let $2 \leq |S_0| \leq 2\alpha - 2$. Assume for contradiction that $|S_{\leq 1}| > 4\alpha - 2$. We show that $F[S_1]$ contains a matching with at least $\alpha$ edges. Applying Observation 2.1 (1) shows that there is a cycle $C$ in $F[S_{\leq 1}]$. Since $F$ is an $S$-forest, this is not an $S$-cycle, so all vertices contained in $C$ are vertices of $S_1$. Let $V_M = \emptyset$. Iteratively adding the two endpoints of an edge of $C$ to $V_M$ and applying Observation 2.1 (1) to $F - V_M$ as long as $|S_{\leq 1} \setminus V_M| > 2\alpha$, we identify $\alpha$ edges of $S_1$ such that all their endpoints are distinct. Thus, $F[S_1]$, and in particular $F[V_M]$, contains a matching $M$ with at least $\alpha$ edges.

Let $C_1, \ldots, C_d$ be the connected components of $F[S_0]$. Notice that $d \leq \alpha$ because $F[S_0]$ is an induced subgraph of a graph with maximum independent set size $\alpha$. By construction, every vertex of $S_1$ is adjacent to at least one vertex of $S_0$. If the endpoints of an edge of $M$ in $S_1$ are adjacent to vertices of the same component $C_i$, $1 \leq i \leq d$, then there is an $S$-cycle in $F$ since every vertex of $C_i$ belongs to $S$. Thus the endpoints of every edge of $M$ are adjacent to different connected components of $F[S_0]$. Now obtain a bipartite graph by contracting every component $C_i$ into a single vertex and every edge of $M$ into a single vertex and keep only the adjacencies between the components and the edges of $M$. Let $(A, B)$ be the bipartition of the resulting bipartite graph such that $A$ contains the components of $F[S_0]$ and $B$ contains the edges of $M$. Since $|A| \leq |B|$ and every vertex of $B$ is adjacent to at least two vertices of $A$, there is a cycle in the bipartite graph. Then, it is not difficult to see that the cycle of the contracted vertices corresponds to an $S$-cycle in $F$. Therefore there is an $S$-cycle in an $S$-forest, leading to a contradiction.

Case 2. Let $2\alpha - 1 \leq |S_0| \leq 2\alpha$. Assume for contradiction that $|S_{\leq 1}| > 2\alpha$. We pick a subset $W$ of $S_1$ such that $|S_0| + |W| = 2\alpha + 1$. Notice that $1 \leq |W| \leq 2$. Then Observation 2.1 (1) implies that there is a cycle in $F[S_0 \cup W]$. Since $W$ has at most two vertices, we conclude that the induced cycle of $F[S_0 \cup W]$ has at least one vertex from $S$, hence it is an $S$-cycle in $F$. Therefore, we reach a contradiction which implies that $|S_{\leq 1}| \leq 2\alpha$. \hfill \qed

Lemma 3.2 shows that we can compute all possible candidates for $S_{\leq 1}$ in polynomial time as follows.

- We first construct, by brute force, all subsets $S'$ of $S$ having at most $2\alpha$ vertices, according to Observation 2.1 (2).

- Then, for each such subset $S'$, we incorporate a set $Y \subseteq N(S') \setminus S$ for which either $|S'| + |Y| \leq 4\alpha - 2$, or $|S'| + |Y| \leq 2\alpha$, according to Lemma 3.2.

- Given the described sets $S'$ and $Y$, we check if $G[S' \cup Y]$ induces an $S$-forest and, if so, we include $S' \cup Y$ into a list $L_1$ containing all candidates for $(S_{\leq 1})$.

The correctness follows from Observation 2.1 and Lemma 3.2. Regarding the running time, notice that we create at most $n^{O(\alpha)}$ subsets for each of $S'$ and $Y \subseteq N(S') \setminus S$. Thus, in $n^{O(\alpha)}$ time we can compute a list $L_1$ that contains all possible candidates for (the solution's) set $S_{\leq 1}$.

Let $S_{\leq 1}$ be a set of $L_1$. We now focus on the graph $G' = G - (S_{\leq 1} \cup S)$. Recall that we assume $S_0 \neq \emptyset$, by the discussion prior to Lemma 3.2. Let $d$ be the number of connected components of $G'$. It is clear that $d \leq \alpha$. In fact, since $|S_0| \geq 1$ and the vertices of $G'$ are at distance of at least two from $S_0$, we have $d < \alpha$.

By brute force, we find all tuples $(A_1, \ldots, A_d)$ such that the following hold:

(i) $A_i \subseteq S_1$, for every $1 \leq i \leq d$, and
(ii) the graph \( \text{aux}_{A_1, \ldots, A_d}(S \leq 1) \) is an \( S \)-forest.

Notice that by the proof of Lemma 3.1 (ii) it is sufficient to consider only such tuples. Since \( A_i \subseteq S \leq 1, d < \alpha, \) and \( |S \leq 1| \leq 4\alpha, \) the number of tuples is \( O(\alpha) \), so that we can obtain the desired set of tuples that satisfy both conditions in polynomial time.

In what follows, we consider the case for \( \alpha \leq 3. \) By the previous arguments, we are given a set \( S \leq 1 \subseteq N[S] \) and tuples of the form \( A_1 \) or \( (A_1, A_2) \) which are subsets of \( S \). Our task is to compute a subset \( V' \) of the vertices of \( G' \) such that the vertices of \( S \leq 1 \cup V' \) induce a maximum \( S \)-forest and the cut of \( (G[S \leq 1], G[V']) \) satisfies \( A_1 \) or \( (A_1, A_2) \), respectively. We distinguish the two cases with the following two lemmas.

**Lemma 3.3.** Let \( X \subseteq N[S] \) and let \( A_1 \) be a subset of \( X \setminus S \) such that both \( G[X] \) and \( \text{aux}_{A_1}(X) \) are \( S \)-forests. There exists a polynomial-time algorithm that computes a maximum \( S \)-forest \( F \) such that \( S \leq 1 = X \) and the cut of its \( S \)-distance partition \( (F \leq 1, F > 1) \) satisfies \( A_1 \).

**Proof.** Since \( F \leq 1 \) is a fixed \( S \)-forest of \( F \), we need to determine the vertices of \( V \setminus (X \cup S) \) that are included in \( F > 1 \). By the desired cut of \( (F \leq 1, F > 1) \), we are restricted to the vertices of \( V \setminus (X \cup S) \) whose neighbors in \( F \leq 1 \) are only vertices of \( A_1 \). Those vertices can be described as follows:

\[
B_1 = \{ w \in V \setminus (X \cup S) \mid N(w) \cap S \leq 1 \subseteq A_1 \}.
\]

Since the cut satisfies a single subset \( A_1 \), we have at most one connected component of \( G[B_1] \) in \( F > 1 \). In order to choose the correct connected component of \( G[B_1] \), we try to include each of them in \( F > 1 \) and select the one having the maximum total weight. Notice that adding any component of \( G[B_1] \) into \( F > 1 \) cannot create any \( S \)-cycle, because \( \text{aux}_{A_1}(X) \) is an \( S \)-forest. Thus, by Lemma 3.1, we correctly compute a maximum \( S \)-forest with the desired properties. Clearly the set \( B_1 \) can be constructed in polynomial time. Since the number of connected components \( G[B_1] \) is at most two, all steps can be executed in polynomial time. \( \square \)

Next, we consider the case when we have a tuple \( (A_1, A_2) \).

**Lemma 3.4.** Let \( X \subseteq N[S] \) and let \( A_1, A_2 \) be subsets of \( X \setminus S \) such that both \( G[X] \) and \( \text{aux}_{A_1, A_2}(X) \) are \( S \)-forests. There exists a polynomial-time algorithm that computes a maximum \( S \)-forest \( F \) such that \( S \leq 1 = X \) and the cut of its \( S \)-distance partition \( (F \leq 1, F > 1) \) satisfies \( (A_1, A_2) \).

**Proof.** Similar to the proof of Lemma 3.3, we first construct the sets \( B_1 \) and \( B_2 \) that contain all vertices of \( V \setminus (X \cup S) \) whose neighbors in \( F \leq 1 \) are only vertices of \( A_1 \) and \( A_2 \) respectively:

\[
B_1 = \{ w \in V \setminus (X \cup S) \mid N(w) \cap S \leq 1 \subseteq A_1 \} \quad \text{and} \quad B_2 = \{ w \in V \setminus (X \cup S) \mid N(w) \cap S \leq 1 \subseteq A_2 \}.
\]

As the desired cut of \( (F \leq 1, F > 1) \) satisfies \( (A_1, A_2) \), there are two connected components of \( F > 1 \) which are subsets of the two sets \( B_1 \) and \( B_2 \), respectively. Let \( C_1 \) and \( C_2 \) be the connected components of \( F > 1 \) such that \( C_1 \subseteq B_1 \) and \( C_2 \subseteq B_2 \). Now observe that there should be two non-adjacent vertices \( c_1 \in B_1 \) and \( c_2 \in B_2 \) that belong to \( C_1 \) and \( C_2 \), respectively. We iterate over all possible pairs of non-adjacent vertices \( c_1 \in B_1 \cap C_1 \) and \( c_2 \in B_2 \cap C_2 \) in \( O(n^2) \) time. Assuming a given choice for \( c_1 \) and \( c_2 \), observe the following:

- Since \( c_1 \) and \( c_2 \) are vertices of different connected components of \( F > 1 \), the components themselves are further restricted to be subsets of \( B_1 \setminus N[c_2] \) and \( B_2 \setminus N[c_1] \), respectively. That is, \( C_1 \subseteq (B_1 \setminus N[c_2]) \) and \( C_2 \subseteq (B_2 \setminus N[c_1]) \).

\[
9
\]
Since $F$ has at least one vertex of $S$, $c_1, c_2 \in V \setminus (X \cup S)$ are non-adjacent, and by the fact $d \leq 3$, we have that $B_1 \setminus N[c_2]$ and $B_2 \setminus N[c_1]$ induce cliques in $G$. Thus $B_1 \setminus N[c_2] \subseteq N[c_1]$ and $B_2 \setminus N[c_1] \subseteq N[c_2]$, respectively.

Then by the second statement it is not difficult to see that $B_1 \setminus N[c_2]$ and $B_2 \setminus N[c_1]$ are disjoint. Let $B'_1 = (B_1 \setminus N[c_2]) \setminus \{c_1\}$ and $B'_2 = (B_2 \setminus N[c_1]) \setminus \{c_2\}$. Now in order to find the maximum induced $S$-forest under the stated conditions and our assumption that $c_1$ and $c_2$ belong to the two connected components of $F_{\geq 1}$, it suffices to find the maximum subset $C_1 \cup C_2$ of $B'_1 \cup B'_2$ such that there are no edges between the vertices of $C_1 \cap B'_1$ and the vertices of $C_2 \cap B'_2$. This boils down to computing a minimum weighted vertex cover on the bipartite graph $G'$ obtained from $G[B'_1 \cup B'_2]$ and removing the edges inside $G[B'_1]$ and $G[B'_2]$. By standard techniques using maximum flow arguments, we compute a minimum weighted vertex cover $U$ on $G'$ in polynomial time [30, 31]. Therefore, $G[B'_1 \cup B'_2] - U$ contains the connected components $C_1 \setminus \{c_1\}$ and $C_1 \setminus \{c_2\}$, as required.

Now we are equipped with the necessary tools in order to obtain our main result, namely a polynomial-time algorithm that solves Weighted Subset Feedback Vertex Set on graphs of independent set number at most 3.

**Theorem 3.5. Weighted Subset Feedback Vertex Set on graphs of independent set number at most 3 can be solved in time $n^{O(1)}$.**

**Proof.** Let us briefly explain such an algorithm for computing a maximum $S$-forest $F$ of a graph $G$ having independent set size at most three. Initially we set $F^* = G - S$. Then, for every set $X \subseteq N[S]$ with $|X| \leq 4 \cdot 3$ such that $G[X]$ is an $S$-forest, we try by brute force every tuple $A_1$ and $(A_1, A_2)$ with $A_i \subseteq (X \setminus S)$ and check whether $\text{aux}_{A_1}(X)$ or $\text{aux}_{A_1, A_2}(X)$ is an $S$-forest. For each of such subsets, we find a maximum $S$-forest $F$ with an $S$-distance partition $(G[X], F_{\geq 1})$ having a cut satisfying $A_1$ or $(A_1, A_2)$, respectively, by applying the algorithms described in Lemma 3.3 and Lemma 3.4. At each step, we maintain the maximum weighted $S$-forest $F^*$ by comparing $F$ with $F^*$. Finally we provide the vertices $V \setminus V(F^*)$ as the set with the minimum total weight that are removed from $G$.

By Lemma 3.2, it is sufficient to consider the described subsets $X$. Since every induced subgraph of $G - X$ contains at most two connected components, Lemma 3.1 implies that all possible subsets $A_1$ or $(A_1, A_2)$ with the described properties are enough to consider. Thus, the correctness follows from Lemmata 3.2–3.4. Regarding the running time, notice that whether a graph contains an $S$-cycle can be tested in polynomial time. Thus, we can construct all described and valid subsets in $n^{O(1)}$ time. Therefore the total running time of the algorithm is $n^{O(1)}$, since each of the algorithms given in Lemma 3.3 and Lemma 3.4, respectively, requires polynomial time.

Let us now show that extending Theorem 3.5 to graphs of larger independent sets is not possible. More precisely, with the following result we show that Weighted Subset Feedback Vertex Set is para-NP-complete parameterized by $\alpha$.

**Theorem 3.6. Weighted Subset Feedback Vertex Set is NP-complete on graphs of independent set number 4.**

**Proof.** We will provide a polynomial reduction from the Vertex Cover problem on tripartite graphs which is NP-complete [18] and asks whether a tripartite graph $G$ contains a vertex cover of size at most $k$. Let $G = (A, B, C, E)$ be a tripartite graph on $n$ vertices, where $(A, B, C)$ is the partition of $V(G)$. We construct a weighted graph $G'$ from $G$ in polynomial time as follows.

- We assign to all vertices of $G$ unary weight.
• We add a vertex \( s \) and the vertices \( u_I, v_I \) for \( I \in \{A, B, C\} \) and we assign weight \( n \) to all those vertices.

• We add all necessary edges to turn each set \( \{u_I, v_I\} \cup I \) into a clique, for \( I \in \{A, B, C\} \), and we make \( s \) adjacent to each vertex \( u_I \), for \( I \in \{A, B, C\} \).

This completes the construction of \( G' \). It is not difficult to verify that the constructed graph \( G' \) is a graph having independent set number 4, since the vertices \( \{s, v_A, v_B, v_C\} \) are pairwise non-adjacent and the vertex set of \( G' - \{s\} \) can be partitioned into three cliques.

Next we claim that \( G \) has a vertex cover \( U \) of size at most \( k < n \) if and only if \( G' \) with \( S = \{s\} \) has a subset feedback vertex set of weight at most \( k \). Assume a vertex cover \( U \) of \( G \). By definition, \( U \) covers all edges of \( G \), so that \( G - U \) is an independent set. It follows that \( \{u_A, v_A\} \cup (A \setminus U), \{u_B, v_B\} \cup (B \setminus U) \) and \( \{u_C, v_C\} \cup (C \setminus U) \) are the connected components of \( G' - (\{s\} \cup U) \). Since \( s \) is only adjacent to the vertices \( u_A, u_B, \) and \( u_C \), no vertex set containing \( s \) induces a cycle of \( G' - U \). Thus, \( G' - U \) is a connected \( S \)-forest. Therefore \( U \) is a subset feedback vertex set of \( (G', \{s\}) \) of weight at most \( k \), because all vertices of \( G \) have unary weight in \( G' \).

For the opposite direction, assume a subset feedback vertex set \( U \) of \( (G', \{s\}) \) having weight at most \( k < n \). If \( U \) is not a subset of \( A \cup B \cup C \), then its sum of weights is greater or equal to \( n \). Thus \( U \) is indeed a subset of \( A \cup B \cup C \). Assume that \( U \) is not a vertex cover of \( G \). By definition, there is an edge of \( G \) that remains uncovered. Without loss of generality, assume that this edge has its endpoints on the vertices \( x \in A \) and \( y \in B \). Then \( \{s, u_A, x, y, u_B\} \) is an induced cycle of \( G' \), which contradicts the fact that \( U \) is a subset feedback vertex set of \( (G', \{s\}) \). Therefore \( U \) is a vertex cover of \( G \) of size at most \( k \), because all vertices of \( G \) have unary weight in \( G' \).

We stress that Theorem 3.6 further implies that the NP-completeness result carries along to graphs of clique cover number four, since the constructed graph given in the proof can be partitioned into four disjoint cliques.

**Corollary 3.7.** **Weighted Subset Feedback Vertex Set** is NP-complete on graphs of clique cover number 4.

Moreover, it is not difficult to generalize the reduction given in Theorem 3.6 on graphs with larger independent set number.

**Theorem 3.8.** **Weighted Subset Feedback Vertex Set** is NP-complete on graphs of independent set number at least 4.

**Proof.** Let \( \alpha \geq 4 \). We apply the same reduction as in the proof of Theorem 3.6. In particular, the reduction comes from the Vertex Cover problem on \((\alpha - 1)\)-partite graphs which is NP-complete for any \( \alpha \geq 4 \) \([18]\). Let \( G = (A_1, \ldots, A_{\alpha - 1}, E) \) be an \((\alpha - 1)\)-partite graph on \( n \) vertices, where \( A_1, \ldots, A_{\alpha - 1} \) is the partition of \( V(G) \). By constructing the graph \( G' \) as explained in Theorem 3.6, we conclude the reduction on the \( \alpha - 1 \) sets \( A_1, \ldots, A_{\alpha - 1} \), instead of the 3 sets \( A, B, C \), for any value of \( \alpha \geq 4 \). Therefore, \( G \) has a vertex cover of size at most \( k < n \) if and only if \( G' \) with \( \alpha(G') = \alpha \) and \( S = \{s\} \) has a subset feedback vertex set of weight at most \( k \). \( \square \)

## 4 SFVS on Graphs of Bounded Independent Set Size

Here we show that despite the complexity dichotomy result for the Weighted Subset Feedback Vertex Set, whenever the weights of the vertices are equal Subset Feedback Vertex Set can be solved in polynomial time on graphs of bounded independent set number.
**Theorem 4.1.** **Subset Feedback Vertex Set** on graphs of independent set number $\alpha$ can be solved in time $n^{O(\alpha)}$.

**Proof.** Let $G = (V, E)$ be a graph with $\alpha = \alpha(G)$ and let $S \subseteq V$. Consider a set $X \subseteq V$ that is a minimum subset feedback vertex set of $G$. Then $F = G - X$ is a maximum $S$-forest of $G$. By Observation 2.1 (2), the $S$-vertices of $F$ are at most $2\alpha$. Thus the set $S \setminus X$ contains at most $2\alpha$ vertices. We now claim that the set $X \setminus S$ also contains at most $2\alpha$ vertices. To see this, observe that if $X \setminus S$ was to contain more than $2\alpha$ vertices, then $S$ would be a subset feedback vertex set of $G$ containing fewer vertices than $X$, leading to a contradiction to the optimality of $X$, hence $X \setminus S$ must contain at most $2\alpha$ vertices.

We conclude that in order to find such a set $X$, it suffices to consider all sets $S' \subseteq S$ and $X' \subseteq V \setminus S$ containing at most $2\alpha$ vertices as candidates for $S \setminus X$ and $X \setminus S$ respectively. To see this, notice that $X \cap S = S \setminus (S \setminus X)$ and $X = (X \cap S) \cup (X \setminus S)$. Moreover, checking whether an induced subgraph of $G$ consists an $S$-forest takes $O(n + m)$ time. Since the number of such sets $S'$ and $X'$ is at most $2n^{2\alpha}$, the total running time is bounded by $n^{O(\alpha)}$. Therefore, in $n^{O(\alpha)}$ time we compute a minimum subset feedback vertex set showing the claimed result.

Regarding the dependence of the exponent in the running time of the algorithm given in Theorem 4.1, note that we can hardly avoid this fact, since Feedback Vertex Set is $W[1]$-hard parameterized by the independent set number as explicitly given in [24]. At the same time such an observation follows from the construction given in [22] in order to prove that Feedback Vertex Set is $W[1]$-hard parameterized by the maximum induced matching width. In the following result, we provide a different and simpler reduction from the Multicolored Independent Set problem [15, 34] which shows an interesting connection with graphs of bounded independent set size.

**Theorem 4.2.** **Feedback Vertex Set** is $W[1]$-hard when parameterized by the clique cover number.

**Proof.** The reduction comes from the Multicolored Independent Set problem: given a graph $G$ and a partition $(V_1, \ldots, V_k)$ of $V(G)$, decide whether $G$ has an independent set of size $k$ containing exactly one vertex from each $V_i$. We call such a set a multicolor independent set of $G$. It is known that Multicolored Independent Set is $W[1]$-hard parameterized by $k$ [15, 34]. Let $(G, V_1, \ldots, V_k)$ be an instance of Multicolored Independent Set where $n = |V(G)|$. From $G$ we construct a graph $H$ as follows.

- We make every set $V_i$ clique by adding all necessary edges.
- For each $V_i$ we add two vertices $x_i, y_i$ that are adjacent to every vertex of $V_i$.
- We add a vertex $z$ that is adjacent to all the vertices of $G$.

This completes the construction of $H$. Observe that $|V(H)| = n + 2k + 1$. Let $X = \{x_1, \ldots, x_k\}$ and $Y = \{y_1, \ldots, y_k\}$. Then $X \cup Y \cup \{z\}$ forms an independent set in $H$ of size $2k + 1$. Notice also that the vertices of each $V_i \cup \{x_i\}$ induce a clique, so that $V(H)$ can be partitioned into $2k + 1$ cliques. Thus, the clique cover number of $H$ is at most $2k + 1$ which implies that $H$ has a clique cover number that is linearly dependent on $k$. We claim that $G$ has a multicolored independent set if and only if $H$ has a feedback vertex set of size at most $n - k$.

Let $I = \{v_1, \ldots, v_k\}$ be a multicolor independent set of $G$ where $v_i \in V_i$ for each $V_i$. We describe an induced forest $F$ of $H$ that contains $3k + 1$ vertices starting from the vertices of $I$. For each vertex $v_i$, we add in $F$ both vertices $x_i$ and $y_i$. Notice that $F$ thus far
Therefore, the $z$ are pairwise non-adjacent, because $F$ is an induced forest with $3k + 1$ vertices, so that $V(H) \setminus V(F)$ constitutes a feedback vertex set of size $n - k$.

For the opposite direction, let $U$ be a feedback vertex set of size $n - k$. Then $F = H - U$ is an induced forest of $H$ that has at least $3k + 1$ vertices. We claim that from each $V_i$ there is exactly one vertex in $F$. Observe that, since each $V_i$ is a clique in $H$, there are at most two vertices from $V_i$ in $F$. We consider the following cases and recall that both $x_i, y_i$ are only adjacent to all the vertices of $V_i$ in $H$:

- If $|V_i \cap V(F)| = 0$ then the vertices of $\{x_i, y_i\} \cup V(F)$ induce a forest, since $X \cup Y$ is an independent set in $H$.
- If $|V_i \cap V(F)| = 1$ then the vertices of $\{x_i, y_i\} \cup V(F)$ induce a forest, since $X \cup Y$ is an independent set in $H$.
- If $|V_i \cap V(F)| = 2$ then $x_i, y_i \notin V(F)$, since both $x_i, y_i$ are adjacent to the vertices of $V_i \cap V(F)$.

This means that $|(V_i \cup \{x_i, y_i\}) \cap V(F)| \leq 2$ whenever $|V_i \cap V(F)| \neq 1$, whereas we have $|(V_i \cup \{x_i, y_i\}) \cap V(F)| \leq 3$ whenever $|V_i \cap V(F)| = 1$. Thus, if there is a set $V_i$ such that $|V_i \cap F| \neq 1$, then

$$|V(F) \setminus \{z\}| = \sum_{i=1}^{k} |V_i \cap V(F)| + |(X \cup Y) \cap V(F)| = \sum_{i=1}^{k} |(V_i \cup \{x_i, y_i\}) \cap V(F)| < 3k.$$ 

This, however, shows that $|V(F)| < 3k + 1$, contradicting the fact that $|V(F)| \geq 3k + 1$, which implies that $|V_i \cap V(F)| = 1$ for every $V_i$. In particular, we have $|V(F) \cap V(G)| = k$. Now observe that $X \cup Y$ is an independent set in $H$ and no vertex from $X \cup Y$ induces a cycle with the vertices from $F$, so that $(X \cup Y) \subseteq V(F)$. Also, notice that if $z \notin V(F)$ then $|V(F)| \leq 3k$. Hence, we conclude that $z \notin V(F)$. Moreover, the vertices of $V(G) \cap V(F)$ are pairwise non-adjacent, because $z$ is a vertex of $F$ and $z$ is adjacent to every vertex of $G$. Therefore, the $k$ vertices of each of $V_i \cap V(F)$ form an independent set in $G$.

Chen et al. [4] showed that $k$-MULTICOLORED INDEPENDENT SET admits no $f(k) \cdot n^{o(k)}$ time algorithm, under the Exponential Time Hypothesis (ETH). Notice that the reduction provided in the proof of Theorem 4.2 is linear in the parameter $k$. Thus we get the following result, conditioned on ETH:

**Corollary 4.3.** FEEDBACK VERTEX SET on graphs of clique cover number $k$ cannot be solved in $f(k) \cdot n^{o(k)}$ time, unless ETH fails.

It should be noted that the stated lower bound shows that the running time of our algorithm given in Theorem 4.1 achieves a tight lower bound.

**Corollary 4.4.** FEEDBACK VERTEX SET on graphs of clique cover number $k$ can be solved in time $n^{O(k)}$.

**Proof.** Let $G$ be a graph of independent set number $\alpha$ and clique cover number $k$. Recall that $k \geq \alpha$. Thus by Theorem 4.1 we can solve SUBSET FEEDBACK VERTEX SET on $G$ in time $n^{O(k)}$. By choosing $S = V(G)$, any $S$-forest is simply a forest of $G$. Therefore we get the claimed result. \qed
5 Extending to other Terminal Set Problems

Let us now consider further terminal set problems that are related to SUBSET FEEDBACK VERTEX SET. In this class of problems we are given a graph \( G = (V,E) \), a terminal set \( T \subseteq V \), and a nonnegative integer \( k \) and the goal is to find a set \( X \subseteq V \) with \( |X| \leq k \) which intersects all “structures” (such as cycles or paths) passing through the vertices in \( T \) [7]. In this setting, SUBSET FEEDBACK VERTEX SET is a particular terminal set problem where the structures to hit are cycles. We show that the ideas that we developed for SUBSET FEEDBACK VERTEX SET on graphs of bounded independent set size can be extended to further terminal set problems where the structures to hit are paths between two terminals instead of cycles containing a terminal.

The (unweighted) NODE MULTIWAY CUT problem is concerned with finding a set \( X \subseteq V \) of size at most \( k \) such that any path between terminals intersects \( X \). It can be formulated as follows.

**NODE MULTIWAY CUT**

*Input:* A graph \( G \), a set \( T \subseteq V \) of terminals, and a nonnegative integer \( k \).

*Task:* Decide whether there is a set \( X \subseteq V \setminus T \) with \( |X| \leq k \) such that any path between two different terminals intersects \( X \).

Notice that in this problem we are not allowed to remove any terminal. For graphs having bounded independent set size, we completely characterize the complexity of NODE MULTIWAY CUT. In particular, for graphs of independent set number at most three we can adapt the reduction given in Theorem 3.6.

**Theorem 5.1.** **NODE MULTIWAY CUT** can be solved on graphs of independent set number at most 2 in \( n^{O(1)} \) time. Moreover, NODE MULTIWAY CUT is NP-complete on graphs of independent set number at least 3.

**Proof.** Let \((G,T,k)\) be an instance of NODE MULTIWAY CUT. If \( G[T] \) contains an edge then we conclude that \((G,T,k)\) is a no-instance, since we are not allowed to remove any vertex from \( T \). In what follows we assume that \( G[T] \) is an independent set. If \( \alpha(G) \leq 2 \) there are at most two terminals, so that \( |T| = 2 \), and we can solve the problem by standard maximum flow techniques [31].

For \( \alpha(G) \geq 3 \), we give a reduction from the NP-complete VERTEX COVER problem on \( \alpha(G) \)-partite graphs, similar to the one given in Theorem 3.6. We give the concrete reduction for \( \alpha(G) = 3 \) which can be straightforward generalized for \( \alpha(G) > 3 \). Let \( G = (A,B,C,E) \) be a tripartite graph where \((A,B,C)\) is the partition of \( V(G) \). We construct a graph \( G' \) from \( G \) by making the three independent sets \( A, B \) and \( C \) into cliques and adding three new vertices \( t_A, t_B, t_C \), that are adjacent to every vertex of \( A, B, \) and \( C \), respectively. It is clear that \( G' \) has independent set size 3. We let \( T = \{t_A,t_B,t_C\} \) and claim that \( G \) has a vertex cover \( U \) of size at most \( k \) if and only if \( G' \) has a set \( X \) of size at most \( k \) which intersects every path between the vertices of \( T \). Removing a vertex cover \( U \) from \( G \) results in a vertex-disjoint union of three cliques in \( G' \) in which each of the vertices \( t_A, t_B, t_C \) belongs to a separate clique. Thus \( X = U \) is a solution for NODE MULTIWAY CUT on \( G' \). For the opposite direction, observe that \( X \) cannot contain any of the three vertices \( t_A, t_B, t_C \). Assume that \( X \) is not a vertex cover of \( G \). Then there is an edge \( \{a,b\} \) that is not covered by \( X \) where \( a \) and \( b \) belong to different partitions of \( V(G) \). Let \( t_a \) and \( t_b \) be the terminal vertices of \( \{t_A,t_B,t_C\} \) which are adjacent to \( a \) and \( b \), respectively, in \( G' \). Then it is clear that there is a path between the terminals \( t_a \) and \( t_b \) in \( G' - X \), leading to a contradiction. Therefore, \( X \) is a vertex cover of \( G \) of size at most \( k \).
Due to the difficulty of Node Multiway Cut even for the unweighted version and with a small size of independent set, we consider a relaxed variation in which we are allowed to remove terminal vertices.

**Algorithm 5.1. (Weighted) Node Multiway Cut with Deletable Terminals**

**Input:** A (vertex-weighted) graph \( G \), a set \( T \subseteq V \) of terminals, and a nonnegative integer \( k \).

**Task:** Decide whether there is a set \( X \subseteq V \) with \( |X| \leq k \) \((w(X) \leq k)\) such that any path between two different terminals intersects \( X \).

Next, we show that the (unweighted) Node Multiway Cut with Deletable Terminals problem can be solved in polynomial time on graphs of bounded independent set number, using an idea similar to the one given in Theorem 4.1.

**Theorem 5.2.** Node Multiway Cut with Deletable Terminals on graphs of independent set number \( \alpha \) can be solved in time \( n^{O(\alpha)} \).

**Proof.** Let \((G, T, k)\) be an instance of Node Multiway Cut with Deletable Terminals where \( G \) is a graph having independent set size at most \( \alpha \). If \( |T| \leq k \) then removing all vertices of \( T \) results in a trivial solution. In what follows, we assume that \( |T| > k \). Observe that every solution \( X \) has size at most \(|T|\). Assume first that \( |T| \leq \alpha \). Then we can enumerate all subsets having at most \(|T|\) vertices in time \( n^{O(|T|)} \) and pick the smallest subset that separates all terminals. Thus, in time \( n^{O(\alpha)} \) we output a valid solution \( X \), if it exists.

Next assume that \( \alpha < |T| \). We consider the graph \( G[T] \). As an induced subgraph of \( G \), \( G[T] \) has independent set size at most \( \alpha \). Thus, \( G[T] \) contains at least one edge. If neither endpoint of an edge in \( G[T] \) belongs to solution \( X \), then there is a path between terminal vertices. This means that there is a vertex cover \( U \) of \( G[T] \) such that \( U \subseteq X \). To compute such a set \( U \), we enumerate all independent sets \( T' \subseteq T \) of size at most \( \alpha \) in time \( |T|^{O(\alpha)} \) and construct \( U = T \setminus T' \). For each constructed \( U \), we consider the graph \( G' = G - U \) with terminals \( T' \). Since \( T' \) is an independent set in \( G' \), we know that \( |T'| \leq \alpha \). Thus, in time \( n^{O(|T'|)} \) we can compute a set \( X' \) of minimum size such that all terminals of \( G' - X' \) are separated. Therefore, the total running time is bounded by \( |T|^{O(\alpha)} \cdot n^{O(|T'|)} \) which is bounded by \( n^{O(\alpha)} \), because \(|T| \leq n\) and \(|T'| \leq \alpha \).

Let us also stress that we can hardly avoid the dependence of the exponent in the running time given in Theorem 5.2. This comes from the fact that Node Multiway Cut with Deletable Terminals with \( T = V(G) \) is equivalent to asking whether the graph contains an independent set of size at least \( k \). That is, we have to solve the Independent Set which is known to be W[1]-hard parameterized by the size of the independent set [13].

Regarding the node-weighted variant of Node Multiway Cut with Deletable Terminals, we provide a dichotomy result with respect to \( \alpha \). In fact, for \( \alpha \leq 2 \) we can invoke the algorithm for the Weighted Subset Feedback Vertex Set given in Theorem 3.5, by adding a new vertex with a large weight that is adjacent to all terminals. Moreover, due to its close connection to the Node Multiway Cut, for \( \alpha \geq 3 \) we can assign appropriate weights to the terminals in a way that they become undeletable. Both ideas are explained in the proof of the following result.

**Theorem 5.3.** Weighted Node Multiway Cut with Deletable Terminals can be solved on graphs of independent set number at most 2 in \( n^{O(1)} \) time. Moreover, Weighted Node Multiway Cut with Deletable Terminals is NP-complete on graphs of independent set number at least 3.

15
Proof. Let \((G, T)\) be an instance of \textsc{Weighted Node Multiway Cut with Deletable Terminals} and assume that \(\alpha(G) \leq 2\). We create an equivalent instance for the \textsc{Weighted Subset Feedback Vertex Set} problem. Starting from \(G\), we obtain a new graph \(G'\) by adding a vertex \(s\) that is adjacent to all terminals of \(T\) and has \(w(T) + 1\) weight. Since we only added one vertex, the size of a maximum independent set of \(G'\) is at most 3. Next we claim that a subset of \(V(G')\) is a solution for \textsc{Weighted Subset Feedback Vertex Set} on the instance \((G', \{s\})\) if and only if it consists a solution for \textsc{Weighted Node Multiway Cut with Deletable Terminals} on \((G, T)\). Notice that a solution for \textsc{Weighted Subset Feedback Vertex Set} on \((G', \{s\})\) cannot contain the new vertex \(s\) due its assigned weight. Also observe that any cycle in \(G'\) passing through \(s\) corresponds to a path in \(G\) connecting two terminals of \(T\) and vice versa. Thus by running the algorithm of Theorem 3.5 on \((G', \{s\})\), we obtain a solution for \textsc{Weighted Node Multiway Cut with Deletable Terminals} on \((G, T)\) in \(n^{O(1)}\) time.

Now let \((G, T, k)\) be an instance for the (unweighted) \textsc{Node Multiway Cut} and assume that \(\alpha(G) \geq 3\). We assign weight \(n\) to every terminal of \(T\) and unary weight to every other vertex. Thus the solutions on \((G, T, k)\) contain only non-terminal vertices for both the (unweighted) \textsc{Node Multiway Cut} and the \textsc{Weighted Node Multiway Cut with Deletable Terminals} problems which implies that they are equivalent. Therefore the \(\text{NP}\)-completeness of \textsc{Weighted Node Multiway Cut with Deletable Terminals} follows, since the (unweighted) \textsc{Node Multiway Cut} is \(\text{NP}\)-complete on graphs of independent set size at least three by Theorem 5.1.

\(\square\)

6 Concluding Remarks

Despite the fact that the \textsc{Weighted Subset Feedback Vertex Set} is \(\text{NP}\)-complete on graphs with bounded independent set number, it is still interesting to settle the complexity of \textsc{Subset Feedback Vertex Set} on graphs of maximum induced matching width by extending the approach given in [23]. Towards such a direction, Dilworth-\(k\) graphs seem a possible candidate for clarifying the complexity status of \textsc{Subset Feedback Vertex Set} (for an exposition of such parameters, see for e.g. [36]). Moreover, \textsc{Feedback Vertex Set} is known to be polynomial-time solvable on cocomparability graphs [28], and, more generally, on AT-free graphs [26]. To our knowledge, \textsc{Subset Feedback Vertex Set} has not been studied on such graphs, besides the existence of a fast exponential-time algorithm for the unweighted variant of the problem [7]. Concerning such an approach, our results indicate that it is natural and compelling to settle first the unweighted \textsc{Subset Feedback Vertex Set} problem. Furthermore, Theorem 5.1 shows that \textsc{Node Multiway Cut} remains \(\text{NP}\)-complete on graphs having maximum induced matching three. However, on graphs of bounded maximum induced matching the complexity of \textsc{Node Multiway Cut with Deletable Terminals} is still unknown.

Acknowledgements

We thank the reviewers for their valuable comments that helped improve the presentation of the paper.

References

2019.


