Classifying Subset Feedback Vertex Set for H-Free Graphs

Giacomo Paesani[1]✉[1][0000−0002−2383−1339], Daniël Paulusma[2][0000−0001−5945−9287], and Paweł Rzążewski[3,4][0000−0001−7696−3848]⋆

[1] School of Computing, University of Leeds, UK,
g.paesani@leeds.ac.uk
[2] Department of Computer Science, Durham University, UK,
daniel.paulusma@durham.ac.uk
[3] Faculty of Mathematics and Information Science, Warsaw University of Technology, Poland
[4] Faculty of Mathematics, Informatics, and Mechanics, University of Warsaw, Poland
pawel.rzazewski@pw.edu.pl

Abstract. In the Feedback Vertex Set problem, we aim to find a small set S of vertices in a graph intersecting every cycle. The Subset Feedback Vertex Set problem requires S to intersect only those cycles that include a vertex of some specified set T. We also consider the Weighted Subset Feedback Vertex Set problem, where each vertex u has weight $w(u) > 0$ and we ask that S has small weight. By combining known NP-hardness results with new polynomial-time results we prove full complexity dichotomies for Subset Feedback Vertex Set and Weighted Subset Feedback Vertex Set for H-free graphs, that is, graphs that do not contain a graph H as an induced subgraph.

Keywords: feedback vertex set · H-free graph · complexity dichotomy

1 Introduction

In a *graph transversal* problem the aim is to find a small set of vertices within a given graph that must intersect every subgraph that belongs to some specified family of graphs. Apart from the Vertex Cover problem, the Feedback Vertex Set problem is perhaps the best-known graph transversal problem. A vertex subset S is a *feedback vertex set* of a graph G if S intersects every cycle of G. In other words, the graph $G - S$ obtained by deleting all vertices of S is a forest. We can now define the problem:

> Feedback Vertex Set
> *Instance:* a graph G and an integer k.
> *Question:* does G have a feedback vertex set S with $|S| \leq k$?

⋆ Supported by Polish National Science Centre grant no. 2018/31/D/ST6/00062.

The Feedback Vertex Set problem is well-known to be NP-complete even under input restrictions. For example, by Poljak's construction [14], the Feedback Vertex Set problem is NP-complete even for graphs of finite girth at least g (the girth of a graph is the length of its shortest cycle). To give another relevant example, Feedback Vertex Set is also NP-complete for line graphs [10].

In order to understand the computational hardness of Feedback Vertex Set better, other graph classes have been considered as well, in particular those that are closed under vertex deletion. Such graph classes are called *hereditary*. It is readily seen that a graph class $\mathcal{G}$ is hereditary if and only if $\mathcal{G}$ can be characterized by a (possibly infinite) set $\mathcal{F}$ of forbidden induced subgraphs. From a systematic point of view it is natural to first consider the case where $\mathcal{F}$ has size 1, say $\mathcal{F} = \{H\}$ for some graph H. This leads to the notion of H-freeness: a graph G is *H-free* for some graph H if G does not contain H as an *induced* subgraph, that is, G cannot be modified into H by a sequence of vertex deletions.

As Feedback Vertex Set is NP-complete for graphs of finite girth at least g for every $g \geq 1$, it is NP-complete for H-free graphs whenever H has a cycle. As it is NP-complete for line graphs and line graphs are claw-free, Feedback Vertex Set is NP-complete for H-free graphs whenever H has an induced claw (the *claw* is the 4-vertex star). In the remaining cases, the graph H is a *linear forest*, that is, the disjoint union of one or more paths. When H is a linear forest, several positive results are known even for the weighted case. Namely, for a graph G, we can define a *(positive) weighting* as a function $w : V \to \mathbb{Q}^+$. For $v \in V$, $w(v)$ is the *weight* of v, and for $S \subseteq V$, we define the weight $w(S) = \sum_{u \in S} w(u)$ of S as the sum of the weights of the vertices in S. This brings us to the following generalization of Feedback Vertex Set:

Weighted Feedback Vertex Set
Instance: a graph G, a positive vertex weighting w of G and a rational number k.
Question: does G have a feedback vertex set S with $w(S) \leq k$?

Note that if w is a constant weighting function, then we obtain the Feedback Vertex Set problem. We denote the r-vertex path by P_r, and the *disjoint union* of two vertex-disjoint graphs G_1 and G_2 by $G_1 + G_2 = ((V(G_1) \cup V(G_2), E(G_1) \cup E(G_2))$, where we write sG for the disjoint union of s copies of G. It is known that Weighted Feedback Vertex Set is polynomial-time solvable for sP_3-free graphs [11] and P_5-free graphs [1]. The latter result was recently extended to $(sP_1 + P_5)$-free graphs for every $s \geq 0$ [11]. We write $H \subseteq_i G$ to denote that H is an *induced* subgraph of G. We can now summarize all known results [1, 10, 11, 14] as follows.

Theorem 1. (Weighted) Feedback Vertex Set *for the class of H-free graphs is polynomial-time solvable if $H \subseteq_i sP_3$ or $H \subseteq_i sP_1 + P_5$ for some $s \geq 1$, and is* NP*-complete if H is not a linear forest.*

Note that the open cases of Theorem 1 are when H is a linear forest with $P_2 + P_4 \subseteq_i H$ or $P_6 \subseteq_i H$.

The (Weighted) Feedback Vertex Set problem can be further generalized in the following way. Let T be some specified subset of vertices of a graph G. A *T-cycle* of G is a cycle that intersects T. A set $S_T \subseteq V$ is a *T-feedback vertex set* of G if S_T contains at least one vertex of every T-cycle; see also Fig. 1. We now consider the following generalizations of Feedback Vertex Set:

Subset Feedback Vertex Set
Instance: a graph G, a subset $T \subseteq V(G)$ and an integer k.
Question: does G have a T-feedback vertex set S_T with $|S_T| \leq k$?

Weighted Subset Feedback Vertex Set
Instance: a graph G, a subset $T \subseteq V(G)$, a positive vertex weighting w of G and a rational number k.
Question: does G have a T-feedback vertex set S_T with $w(S_T) \leq k$?

The NP-complete cases in Theorem 1 carry over to (Weighted) Subset Feedback Vertex Set; just set $T := V(G)$ in both cases. However, this is no longer true for the polynomial-time cases: Fomin et al. [7] proved NP-completeness of Subset Feedback Vertex Set for split graphs, which form a subclass of $2P_2$-free graphs. Interestingly, Papadopoulos and Tzimas [13] proved that Weighted Subset Feedback Vertex Set is NP-complete for $5P_1$-free graphs, whereas Brettell et al. [4] proved that Subset Feedback Vertex Set can be solved in polynomial time even for $(sP_1 + P_3)$-free graphs for every $s \geq 1$ [4]. Hence, in contrast to many other transversal problems, the complexities on the weighted and unweighted subset versions do not coincide for H-free graphs.

It is also known that Weighted Subset Feedback Vertex Set can be solved in polynomial time for permutation graphs [12] and thus for its subclass of P_4-free graphs. The latter result also follows from a more general result related to the graph parameter mim-width [15]. Namely, Bergougnoux, Papadopoulos and Telle [3] proved that Weighted Subset Feedback Vertex

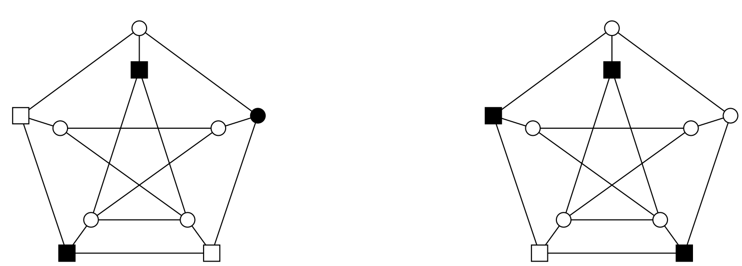

Fig. 1. Two examples of a slightly modified Petersen graph with the set T indicated by square vertices. In both examples, the set S_T of black vertices is a T-feedback vertex set. On the left, $S_T \setminus T \neq \emptyset$. On the right, $S_T \subseteq T$.

SET is polynomial-time solvable for graphs for which we can find a decomposition of constant mim-width in polynomial time [3]; the class of P_4-free graphs is an example of such a class. Brettell et al. [5] extended these results by proving that WEIGHTED SUBSET FEEDBACK VERTEX SET, restricted to H-free graphs, is polynomial-time solvable if $H \subseteq_i 3P_1 + P_2$ or $H \subseteq_i P_1 + P_3$.

The above results leave open a number of unresolved cases for both problems, as identified in [4] and [5], where the following open problems are posed:

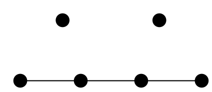

Fig. 2. The graph $2P_1 + P_4$.

Open Problem 1 *Determine the complexity of* WEIGHTED SUBSET FEEDBACK VERTEX SET *for H-free graphs if $H \in \{2P_1 + P_3, P_1 + P_4, 2P_1 + P_4\}$.*

Open Problem 2 *Determine the complexity of* SUBSET FEEDBACK VERTEX SET *for H-free graphs if $H = sP_1 + P_4$ for some integer $s \geq 1$.*

1.1 Our Results

We completely solve Open Problems 1 and 2.

In Section 3, we prove that WEIGHTED SUBSET FEEDBACK VERTEX SET is polynomial-time solvable for $(2P_1 + P_4)$-free graphs. This result generalizes all known polynomial-time results for WEIGHTED FEEDBACK VERTEX SET. It also immediately implies polynomial-time solvability for the other two cases in Open Problem 1, as $(2P_1+P_3)$-free graphs and (P_1+P_4)-free graphs form subclasses of $(2P_1 + P_4)$-free graphs. Combining the aforementioned NP-completeness results of [7] and [13] for $2P_2$-free graphs and $5P_1$-free graphs, respectively, with the NP-completeness results in Theorem 1 for the case where H has a cycle or a claw and this new result gives us the following complexity dichotomy (see also Fig. 2).

Theorem 2. *For a graph H, the* WEIGHTED SUBSET FEEDBACK VERTEX SET *problem on H-free graphs is polynomial-time solvable if $H \subseteq_i 2P_1 + P_4$, and is* NP*-complete otherwise.*

In Section 4 we solve Open Problem 2 by proving that SUBSET FEEDBACK VERTEX SET can be solved in polynomial time for $(sP_1 + P_4)$-free graphs, for every $s \geq 1$. This result generalizes all known polynomial-time results for WEIGHTED FEEDBACK VERTEX SET. After combining it with the aforementioned NP-completeness results of [7] and Theorem 1 we obtain the following complexity dichotomy.

Theorem 3. *For a graph H, the* Subset Feedback Vertex Set *problem on H-free graphs is polynomial-time solvable if $H \subseteq_i sP_1 + P_4$ for some $s \geq 0$, and is* NP*-complete otherwise.*

Due to Theorems 2 and 3 we now know where exactly the complexity jump between the weighted and unweighted versions occurs.

Our proof technique for these results is based on the following two ideas. First, if the complement F_T of a T-feedback vertex set contains s vertices of small degree in F_T, then we can "guess" these vertices and their neighbours in F_T. We then show that after removing all the other neighbours of small-degree vertices, we will obtain a graph of small mim-width. If F_T does not contain s small-degree vertices, we will argue that F_T contains a bounded number of vertices of T. We guess these vertices and exploit their presence. This is straightforward for Subset Feedback Vertex Set but more involved for Weighted Subset Feedback Vertex Set. The latter was to be expected from the hardness construction for Weighted Subset Feedback Vertex Set on $5P_1$-free graphs, in which $|T| = 1$ (but as we will show our algorithm is able to deal with that construction due to the $(2P_1 + P_4)$-freeness of the input graph).

We finish our paper with a brief discussion on related graph transversal problems and some open questions in Section 5.

2 Preliminaries

Let $G = (V, E)$ be a graph. If $S \subseteq V$, then $G[S]$ denotes the subgraph of G induced by S, and $G - S$ is the graph $G[V \setminus S]$. We say that S is *independent* if $G[S]$ has no edges, and that S is a *clique* and $G[S]$ is *complete* if every pair of vertices in S is joined by an edge. A *(connected) component* of G is a maximal connected subgraph of G. The *neighbourhood* of a vertex $u \in V$ is the set $N(u) = \{v \mid uv \in E\}$. A graph is *bipartite* if its vertex set can be partitioned into at most two independent sets.

Recall that for a graph $G = (V, E)$ and a subset $T \subseteq V$, a T-feedback vertex set is a set $S \subseteq V$ that intersects all T-cycles. Note that $G - S$ is a graph that has no T-cycles; we call such a graph a *T-forest*. Thus the problem of finding a T-feedback vertex set of minimum size is equivalent to finding a T-forest of maximum size. Similarly, the problem of finding a T-feedback vertex set of minimum weight is equivalent to finding a T-forest of maximum weight. These maximisation problems are actually the problems that we will solve. Consequently, any T-forest will be called a *solution* for an instance (G, T) or (G, w, T), respectively, and our aim is to find a solution of maximum size or maximum weight, respectively.

Throughout our proofs we will need to check if some graph F is a solution. The following lemma shows that we can recognize solutions in linear time. The lemma combines results claimed but not proved in [9, 13]. It is easy to show but for an explicit proof we refer to [4, Lemma 3].

Lemma 1. *It is possible to decide in $O(n+m)$ time if a graph F is a T-forest for some given set $T \subseteq V(F)$.*

In our proofs we will not refer to Lemma 1 explicitly, but we will use it implicitly every time we must check if some graph F is a solution.

3 The Weighted Variant

In this section, we present our polynomial-time algorithm for Weighted Subset Feedback Vertex Set on $(2P_1 + P_4)$-free graphs.

Outline. Our algorithm is based on the following steps. We first show in Section 3.1 how to compute a solution F that contains at most one vertex from T, which moreover has small degree in F. In Section 3.2 we then show that if two vertices of small degree in a solution are non-adjacent, we can exploit the $(2P_1 + P_4)$-freeness of the input graph G to reduce to a graph G' of bounded mim-width. The latter enables us to apply the algorithm of Bergougnoux, Papadopoulos and Telle [3]. In Section 3.3 we deal with the remaining case, where all the vertices of small degree in a solution F form a clique and F contains at least two vertices of T. We first show that every vertex of T that belongs to F must have small degree in F. Hence, as the vertices in $T \cap V(F)$ must also induce a forest, F has exactly two adjacent vertices of T, each of small degree in F. This structural result enables us to do a small case analysis. We combine this step together with our previous algorithmic procedures into one algorithm.

Remark. Some of the lemmas in the following three subsections hold for $(sP_1 + P_4)$-free graphs, for every $s \geq 2$, or even for general graphs. In order to re-use these lemmas in Section 4, where we consider Subset Feedback Vertex Set for $(sP_1 + P_4)$-free graphs, we formulate these lemmas as general as possible.

3.1 Three Special Types of Solutions

In this section we will show how we can find three special types of solutions in polynomial time for $(2P_1 + P_4)$-free graphs. These solutions have in common that they contain at most one vertex from the set T and moreover, this vertex has small degree in F.

Let $G = (V, E)$ be a graph and let $T \subseteq V$ be a subset of vertices of G. A T-forest F is a $\leq$1-*part solution* if F contains at most one vertex from T and moreover, if F contains a vertex u from T, then u has degree at most 1 in F. The following lemma holds for general graphs and is easy to see.

Lemma 2. *For a graph $G = (V, E)$ with a positive vertex weighting w and a set $T \subseteq V$, it is possible to find a $\leq$1-part solution of maximum weight in polynomial time.*

Let $G = (V, E)$ be a graph and let $T \subseteq V$ be a subset of vertices of G. A T-forest F is a 2-*part solution* if F contains exactly one vertex u of T and u has exactly two neighbours v_1 and v_2 in F. We say that u is the *center* of F and that v_1 and v_2 are the *center neighbours*. Let A be the connected component of F that contains u. Then we say that A is the *center component* of F. We will prove how to find 2-part solutions in polynomial time even for general graphs. In order to do this, we will reduce to a classical problem, namely:

Weighted Vertex Cut
Instance: a graph $G = (V, E)$, two distinct non-adjacent terminals t_1 and t_2, and a positive vertex weighting w.
Task: determine a set $S \subseteq V \setminus \{t_1, t_2\}$ of minimum weight such that t_1 and t_2 are in different connected components of $G - S$.

The Weighted Vertex Cut problem is well known to be polynomial-time solvable by standard network flow techniques.

Lemma 3. Weighted Vertex Cut *is polynomial-time solvable.*

We use Lemma 3 in several of our proofs, including in the (omitted) proof of the next lemma.

Lemma 4. *For a graph $G = (V, E)$ with a positive vertex weighting w and a set $T \subseteq V$, it is possible to find a 2-part solution of maximum weight in polynomial time.*

Let $G = (V, E)$ be a graph and let $T \subseteq V$ be a subset of vertices of G. A T-forest F is a 3*-part solution* if F contains exactly one vertex u of T and u has exactly three neighbours v_1, v_2, v_3 in F. Again we say that u is the *center* of F; that v_1, v_2, v_3 are the *center neighbours*; and that the connected component of F that contains u is the *center component* of F. We can show the following lemma (proof omitted).

Lemma 5. *For a $(2P_1 + P_4)$-free graph $G = (V, E)$ with a positive vertex weighting w and a set $T \subseteq V$, it is possible to find a 3-part solution of maximum weight in polynomial time.*

3.2 Mim-Width

We also need some known results that involve the mim-width of a graph. This width parameter was introduced by Vatshelle [15]. For the definition of mim-width we refer to [15], as we do not need it here. A graph class $\mathcal{G}$ has *bounded* mim-width if there exists a constant c such that every graph in $\mathcal{G}$ has mim-width at most c. The mim-width of a graph class $\mathcal{G}$ is *quickly computable* if it is possible to compute in polynomial time a so-called branch decomposition for a graph $G \in \mathcal{G}$ whose mim-width is bounded by some function in the mim-width of G. We can now state the aforementioned result of Bergougnoux, Papadopoulos and Telle in a more detailed way.

Theorem 4 ([3]). Weighted Subset Feedback Vertex Set *is polynomial-time solvable for every graph class whose mim-width is bounded and quickly computable.*

Belmonte and Vatshelle [2] proved that the mim-width of the class of permutation graphs is bounded and quickly computable. As P_4-free graphs form a subclass of the class of permutation graphs, we immediately obtain the following lemma.[5]

Lemma 6. *The mim-width of the class of P_4-free graphs is bounded and quickly computable.*

For a graph class $\mathcal{G}$ and an integer $p \geq 0$, we let $\mathcal{G} + pv$ be the graph class that consists of all graphs that can be modified into a graph from $\mathcal{G}$ by deleting at most p vertices. The following lemma follows in a straightforward way from a result of Vatshelle [15].

Lemma 7. *If $\mathcal{G}$ is a graph class whose mim-width is bounded and quickly computable, then the same holds for the class $\mathcal{G} + pv$, for every constant $p \geq 0$.*

Let $G = (V, E)$ be an $(sP_1 + P_4)$-free graph for some $s \geq 2$ and let $T \subseteq V$. Let F be a T-forest of G. We define the *core* of F as the set of vertices of F that have at most $2s - 1$ neighbours in F. We say that F is *core-complete* if the core of F has no independent set of size at least s; otherwise F is *core-incomplete*.[6] We use the above results to show the following algorithmic lemma (proof omitted).

Lemma 8. *Let $s \geq 2$. For an $(sP_1 + P_4)$-free graph $G = (V, E)$ with a positive vertex weighting w and a set $T \subseteq V$, it is possible to find a core-incomplete solution of maximum weight in polynomial time.*

3.3 The Algorithm

In this section we present our algorithm for Weighted Subset Feedback Vertex Set restricted to $(2P_1 + P_4)$-free graphs. We first need to prove one more structural lemma for core-complete solutions. We prove this lemma for any value $s \geq 2$, such that we can use this lemma in the next section as well. However, for $s = 2$ we have a more accurate upper bound on the size of the core.

Lemma 9. *For some $s \geq 2$, let $G = (V, E)$ be an $(sP_1 + P_4)$-free graph. Let $T \subseteq V$. Let F be a core-complete T-forest of G such that $T \cap V(F) \neq \emptyset$. Then the core of F contains every vertex of $T \cap V(F)$, and $T \cap V(F)$ has size at most $2s-2$. If $s = 2$, the core of F is a clique of size at most 2 (in this case $T \cap V(F)$ has size at most 2 as well).*

[5] It is well-known that P_4-free graphs have clique-width at most 2, and instead of Theorem 4 we could have used a corresponding result for clique-width. We chose to formulate Theorem 4 in terms of mim-width, as mim-width is a more powerful parameter than clique-width [15] and thus bounded for more graph classes.

[6] These notions are not meaningful if $s \in \{0, 1\}$. Hence, we defined them for $s \geq 2$.

Proof. Consider a vertex $u \in T \cap V(F)$. For a contradiction, assume that u does not belong to the core of F. Then u has at least $2s$ neighbours in F. Let $V_u = \{v_1, \dots, v_p\}$ for some $p \geq 2s$ be the set of neighbours of u in F.

Let A be the connected component of F that contains u. As F is a T-forest, $A - u$ consists of p connected components $D_1, \dots, D_p$ such that $v_i \in V(D_i)$ for $i \in \{1, \dots, p\}$. In particular, this implies that $V_u = \{v_1, \dots, v_p\}$ must be an independent set. As the core of F has no independent set of size s, this means that at most $s-1$ vertices of V_u may belong to the core of F. Recall that $p \geq 2s$. Hence, we may assume without loss of generality that $v_1, \dots, v_{s+1}$ do *not* belong to the core of F. This means that $v_1, \dots, v_{s+1}$ each have degree at least $2s$ in A. Hence, for $i \in \{1, \dots, s+1\}$, vertex v_i is adjacent to some vertex w_i in D_i. As $s \geq 2$, we have that $2s > s+1$ and hence, vertex v_{s+2} exists. However, now the vertices $w_1, v_1, u, v_{s+2}, w_2, w_3, \dots, w_{s+1}$ induce an $sP_1 + P_4$, a contradiction (see also Fig. 3).

From the above, we conclude that every vertex of $T \cap V(F)$ belongs to the core of F. As F is a T-forest, $T \cap V(F)$ induces a forest, and thus a bipartite graph. As F is core-complete, every independent set in the core has size at most $s-1$. Hence, $T \cap V(F)$ has size at most $2(s-1) = 2s-2$.

Now suppose that $s = 2$. As F is core-complete, the core of F must be a clique. As the core of F contains $T \cap V(F)$ and $T \cap V(F)$ induces a forest, this means that the core of F, and thus also $T \cap V(F)$, has size at most 2. This completes the proof of the lemma. □

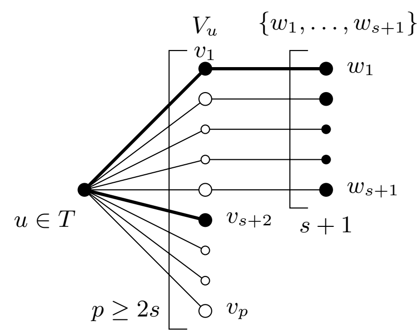

Fig. 3. An example of the contradiction obtained in Lemma 9: the assumption that a vertex $u \in T$ does not belong to the core of a core-complete solution leads to the presence of an induced $sP_1 + P_4$ (highlighted by the black vertices and thick edges).

By using the above results and the results from Sections 3.1 and 3.2, we are now able to prove our main result.

Theorem 5. Weighted Subset Feedback Vertex Set *is polynomial-time solvable for* $(2P_1 + P_4)$*-free graphs.*

Proof. Let $G = (V, E)$ be a $(2P_1 + P_4)$-free graph, and let T be some subset of V. Let w be a positive vertex weighting of G. We aim to find a maximum weight T-forest F for (G, T, w) (recall that we call T-forests solutions for our problem). As $s = 2$, the core of F is, by definition, the set of vertices of F that have maximum degree at most 3 in F.

We first compute a core-incomplete solution of maximum weight; this takes polynomial time by Lemma 8 (in which we set $s = 2$). We will now compute in polynomial time a core-complete solution F of maximum weight for (G, T, w). We then compare the weights of the two solutions found to each other and pick one with the largest weight.

By Lemma 9, it holds for every core-complete solution F that $T \cap V(F)$ belongs to the core of F, and moreover that $|T \cap V(F)| \leq 2$. We first compute a core-complete solution F with $|T \cap V(F)| \leq 1$ of maximum weight. As $T \cap V(F)$ belongs to the core of F, we find that if $|T \cap V(F)| = 1$, say $T \cap V(F) = \{u\}$ for some $u \in T$, then u has maximum degree at most 3 in F. Hence, in the case where $|T \cap V(F)| \leq 1$, it suffices to compute a $\leq$1-part solution, 2-part solution and 3-part solution for (G, T, w) of maximum weight and to remember one with the largest weight. By Lemmas 2, 4 and 5, respectively, this takes polynomial time.

It remains to compute a core-complete solution F with $|T \cap V(F)| = 2$ of maximum weight. By Lemma 9, it holds for every such solution F that both vertices of $T \cap V(F)$ are adjacent and are the only vertices that belong to the core of F.

We consider all $O(n^2)$ possibilities of choosing two adjacent vertices of T to be the two core vertices of F. Consider such a choice of adjacent vertices u_1, u_2. So, u_1 and u_2 are the only vertices of degree at most 3 in the solution F that we are looking for and moreover, all other vertices of T do not belong to F.

Suppose one of the vertices u_1, u_2 has degree 1 in F. First let this vertex be u_1. Then we remove u_1 and all its neighbours except for u_2 from G. Let G' be the resulting graph. Let $T' = T \setminus (\{u_1\} \cup (N(u_1) \setminus \{u_2\}))$, and let w' be the restriction of w to G'. We now compute for (G', w', T'), a $\leq$1-part solution and 2-part solution of maximum weight with u_2 as center. By Lemmas 2 and 4, respectively, this takes polynomial time.[7] We then add u_1 back to the solution to get a solution for (G, w, T). We do the same steps with respect to u_2. In the end we take a solution with largest weight.

So from now on, assume that both u_1 and u_2 have degree at least 2 in F. We first argue that in this case both u_1 and u_2 have degree exactly 2 in F. For a contradiction, suppose u_1 has degree 3 in F (recall that u_1 has degree at most 3 in F). Let v_1 and v_1' be two distinct neighbours of u_1 in $V(F) \setminus \{u_2\}$. Let v_2 be a neighbour of u_2 in $V(F) \setminus \{u_1\}$. As F is a T-forest, v_1, v_1', v_2 belong to distinct connected components D_1, D_1' and D_2, respectively, of $F - \{u_1, u_2\}$. As the core of F consists of u_1 and u_2 only, v_1, v_1', v_2 each have a neighbour x_1, x_1', x_2 in D_1,

[7] Strictly speaking, this statement follows from the proofs of these two lemmas, as we have fixed u_2 as the center.

D_1' and D_2, respectively. However, now $x_2, v_2, u_2, u_1, x_1, x_1'$ induce a $2P_1 + P_4$ in F and thus also in G, a contradiction; see also Fig. 4.

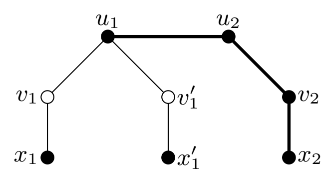

Fig. 4. The situation in Theorem 5 where u_1 has degree at least 3 in F and u_2 has degree 2 in F; this leads to the presence of an induced $2P_1 + P_4$ (highlighted by the black vertices and thick edges).

From the above we conclude that each of u_1 and u_2 has exactly one other neighbour in F. Call these vertices v_1 and v_2, respectively. We consider all $O(n^2)$ possibilities of choosing v_1 and v_2. As F is a T-forest, $G - \{u_1, u_2\}$ consists of two connected components D_1 and D_2, such that v_1 belongs to D_1 and v_2 belongs to D_2.

Let G' be the graph obtained from G by removing every vertex of T, every neighbour of u_1 except v_1 and every neighbour of u_2 except v_2. Let w' be the restriction of w to G'. Then, it remains to solve Weighted Vertex Cut for the instance (G', v_1, v_2, w'). By Lemma 3, this can be done in polynomial time. Out of all the solutions found for different pairs u_1, u_2 we take one with the largest weight. Note that we found this solution in polynomial time, as the number of branches is $O(n^4)$.

As mentioned we take a solution of maximum weight from all the solutions found in the above steps. The correctness of our algorithm follows from the fact that we exhaustively considered all possible situations. Moreover, the number of situations is polynomial and processing each situation takes polynomial time. Hence, the running time of our algorithm is polynomial. □

4 The Unweighted Variant

In this section, we present our polynomial-time algorithm for Subset Feedback Vertex Set on $(sP_1 + P_4)$-free graphs for every $s \geq 0$. As this problem is a special case of Weighted Subset Feedback Vertex Set (namely when $w \equiv 1$), we can use some of the structural results from the previous section.

Theorem 6. Subset Feedback Vertex Set *is polynomial-time solvable on $(sP_1 + P_4)$-free graphs for every $s \geq 0$.*

Proof. Let $G = (V, E)$ be an $(sP_1 + P_4)$-free graph for some integer s, and let $T \subseteq V$. Let $|V| = n$. As the class of $(sP_1 + P_4)$-free graphs is a subclass of the class of $((s+1)P_1 + P_4)$-free graphs, we may impose any lower bound on s; we set $s \geq 2$. We aim to find a T-forest F of G of maximum size (recall that we call T-forests solutions for our problem).

We first compute a maximum-size core-incomplete solution for (G, T). By Lemma 8, this takes polynomial time. It remains to compare the size of this solution with a maximum-size core-complete solution, which we compute below.

By Lemma 9, we find that $T \cap V(F)$ has size at most $2s - 2$ for every core-complete solution F. We consider all $O(n^{2s-2})$ possibilities of choosing the vertices of $T \cap V(F)$. For each choice of $T \cap V(F)$ we do as follows. We note that the set of vertices of $G - T$ that do not belong to F has size at most $|T \cap V(F)|$; otherwise $F' = V \setminus T$ would be a larger solution than F. Hence, we can consider all $O(n^{|T \cap V(F)|}) = O(n^{2s-2})$ possibilities of choosing the set of vertices of $G - T$ that do not belong to F, or equivalently, of choosing the set of vertices of $G - T$ that *do* belong to F. In other words, we guessed F by brute force, and the number of guesses is $O(n^{4s-4})$. In the end we found in polynomial time a maximum-size core-complete solution. We compare it with the maximum-size core-incomplete solution found above and pick one with the largest size. $\square$

5 Conclusions

By combining known hardness results with new polynomial-time results, we completely classified the complexities of Weighted Subset Feedback Vertex Set and Subset Feedback Vertex Set for H-free graphs. We recall that the classical versions Weighted Feedback Vertex Set and Feedback Vertex Set are not yet completely classified (see Theorem 1).

We now briefly discuss the variant where instead of intersecting every T-cycle, a solution only needs to intersect every T-cycle of *odd* length. These two problems are called Weighted Subset Odd Cycle Transversal and Subset Odd Cycle Transversal, respectively. So far, these problems behave in exactly the same way on H-free graphs as their feedback vertex set counterparts (see [4] and [5]). So, the only open cases for Weighted Subset Odd Cycle Transversal on H-free graphs are the ones where $H \in \{2P_1 + P_3, P_1 + P_4, 2P_1 + P_4\}$ and the only open cases for Subset Odd Cycle Transversal on H-free graphs are the ones where $H = sP_1 + P_4$ for some $s \geq 1$. As solutions F for these problems may only contain vertices of T of high degree, we can no longer use our proof technique, and new ideas are needed.

We note, however, that complexity dichotomies of Weighted Subset Odd Cycle Transversal and Subset Odd Cycle Transversal do not have to coincide with those in Theorems 2 and 3 for their feedback vertex set counterparts. After all, the complexities of the corresponding classical versions may not coincide either. Namely, it is known that Odd Cycle Transversal is NP-complete for $(P_2 + P_5, P_6)$-free graphs [6], and thus for $(P_2 + P_5)$-free graphs and P_6-free graphs, whereas for Feedback Vertex Set such a hardness result is unlikely: for every linear forest H, Feedback Vertex Set is quasipolynomial-time solvable on H-free graphs [8].

References

1. Abrishami, T., Chudnovsky, M., Pilipczuk, M., Rzążewski, P., Seymour, P.: Induced subgraphs of bounded treewidth and the container method. Proc. SODA 2021, 1948–1964 (2021)
2. Belmonte, R., Vatshelle, M.: Graph classes with structured neighborhoods and algorithmic applications. Theoretical Computer Science **511**, 54–65 (2013)
3. Bergougnoux, B., Papadopoulos, C., Telle, J.A.: Node Multiway Cut and Subset Feedback Vertex Set on graphs of bounded mim-width. Proc. WG 2020, LNCS **12301**, 388–400 (2020)
4. Brettell, N., Johnson, M., Paesani, G., Paulusma, D.: Computing subset transversals in H-free graphs. Theoretical Computer Science **898**, 59–68 (2022)
5. Brettell, N., Johnson, M., Paulusma, D.: Computing weighted subset transversals in H-free graphs. Journal of Computer and System Sciences **128**, 71–85 (2022)
6. Dabrowski, K.K., Feghali, C., Johnson, M., Paesani, G., Paulusma, D., Rzążewski, P.: On cycle transversals and their connected variants in the absence of a small linear forest. Algorithmica **82**, 2841–2866 (2020)
7. Fomin, F.V., Heggernes, P., Kratsch, D., Papadopoulos, C., Villanger, Y.: Enumerating minimal subset feedback vertex sets. Algorithmica **69**, 216–231 (2014)
8. Gartland, P., Lokshtanov, D., Pilipczuk, M., Pilipczuk, M., Rzążewski, P.: Finding large induced sparse subgraphs in $C_{>t}$-free graphs in quasipolynomial time. Proc. STOC 2021, 330–341 (2021)
9. Lokshtanov, D., Misra, P., Ramanujan, M.S., Saurabh, S.: Hitting selected (odd) cycles. SIAM Journal on Discrete Mathematics, **31**, 1581–1615 (2017)
10. Munaro, A.: On line graphs of subcubic triangle-free graphs. Discrete Mathematics **340**, 1210–1226 (2017)
11. Paesani, G., Paulusma, D., Rzążewski, P.: Feedback Vertex Set and Even Cycle Transversal for H-free graphs: finding large block graphs. SIAM Journal on Discrete Mathematics, to appear.
12. Papadopoulos, C., Tzimas, S.: Polynomial-time algorithms for the subset feedback vertex set problem on interval graphs and permutation graphs. Discrete Applied Mathematics **258**, 204–221 (2019)
13. Papadopoulos, C., Tzimas, S.: Subset feedback vertex set on graphs of bounded independent set size. Theoretical Computer Science **814**, 177–188 (2020)
14. Poljak, S.: A note on stable sets and colorings of graphs. Commentationes Mathematicae Universitatis Carolinae **15**, 307–309 (1974)
15. Vatshelle, M.: New Width Parameters of Graphs. PhD thesis, University of Bergen (2012).