Connected Vertex Cover
for \((sP_1 + P_5)\)-Free Graphs

Matthew Johnson, Giacomo Paesani, and Daniël Paulusma
Department of Computer Science, Durham University, UK
{matthew.johnson2,giacomo.paesani,daniel.paulusma}@durham.ac.uk

Abstract. The **Connected Vertex Cover** problem is to decide if a graph \(G\) has a vertex cover of size at most \(k\) that induces a connected subgraph of \(G\). A graph is \(H\)-free if it does not contain \(H\) as an induced subgraph. We prove that **Connected Vertex Cover** is polynomial-time solvable for \((sP_1 + P_5)\)-free graphs for all \(s \geq 0\).

1 Introduction

A subset \(S\) of vertices of a graph \(G\) is a **vertex cover** of \(G\) if every edge of \(G\) has at least one end-point in \(S\), and \(S\) is an an **independent set** if no two vertices in \(S\) are adjacent. These notions lead to two classical graph problems (which are both \(NP\)-complete). The **Vertex Cover** problem is to decide if a given graph \(G\) has a vertex cover of size at most \(k\) for some given integer \(k\). The **Independent Set** problem is to decide if a given graph \(G\) has an independent set of size at least \(\ell\) for some given integer \(\ell\). A set \(S\) of at least \(k\) vertices of a graph \(G\) on \(n\) vertices is a vertex cover of \(G\) if and only if \(V_G \setminus S\) is an independent set of \(G\) (of size at most \(n - k\)). Hence **Vertex Cover** and **Independent Set** are polynomially equivalent.

A vertex cover of a graph \(G\) is connected if it induces a connected subgraph of \(G\). In our paper we focus on the corresponding decision problem.

**Connected Vertex Cover**

**Instance:** a graph \(G\) and an integer \(k\).

**Question:** does \(G\) have a connected vertex cover \(S\) with \(|S| \leq k\)?

In 1977, Garey and Johnson [16] proved that **Connected Vertex Cover** is \(NP\)-complete even for planar graphs of maximum degree 4. More recently, Priyadarsini and Hemalatha [33] and Fernau and Manlove [14] strengthened this result to 2-connected planar graphs of maximum degree 4 and planar bipartite graphs of maximum degree 4, respectively. Watanabe, Kajita, and Onaga [37] proved that **Connected Vertex Cover** is \(NP\)-complete even for 3-connected graphs, and in 2017, Munaro [31] proved the same for line graphs of planar cubic bipartite graphs and for planar bipartite graphs of arbitrarily large girth.

We now turn to tractable cases. Ueno, Kajitani, and Gotoh [39] proved that **Connected Vertex Cover** is polynomial-time solvable for graphs of maximum degree at most 3 and for trees. Escoffier, Gourvès, and Monnot [13] extended the latter result to chordal graphs. As **Vertex Cover** is also polynomial-time solvable for chordal graphs [17], the same authors proposed a general study on the complexity of **Connected Vertex Cover** on graph classes for which **Vertex Cover** is polynomial-time solvable. This is also the research question of our paper:

---

*This work was supported by The Leverhulme Trust (Grant RPG-2016-258).*
For which classes of graphs do the complexities of Vertex Cover and Connected Vertex Cover coincide?

As a next step in this direction, Chiarelli et al. [10] considered classes of graphs characterized by a single forbidden induced subgraph $H$. Such graphs are also called $H$-free. They observed that the results of Munaro [31] imply that Connected Vertex Cover is NP-complete for $H$-free graphs whenever $H$ contains a cycle or a claw. By using Poljak’s construction [32], Vertex Cover is readily seen to be NP-complete for graphs of arbitrarily large girth and thus for $H$-free graphs whenever $H$ contains a cycle. However, when $H$ is the claw, Vertex Cover becomes polynomial-time solvable for $H$-free graphs [29,33]. Hence, there exists at least one graph $H$ such that Connected Vertex Cover and Vertex Cover have different complexities when restricted to $H$-free graphs (assuming $P \neq NP$).

From the above it follows that the complexity of Connected Vertex Cover has been determined for $H$-free graphs except when $H$ is a linear forest, that is, the disjoint union of one or more paths. Even the case where $H$ is a single path is settled neither for Vertex Cover nor for Connected Vertex Cover. In particular, it is not known if there exists an integer $r$, such that Vertex Cover or Connected Vertex Cover is NP-complete for $P_r$-free graphs (the graph $P_r$ denotes the $r$-vertex path). Lokshin, Vatshelle, and Villanger [28] proved that Independent Set, and thus Vertex Cover, is polynomial-time solvable for $P_5$-free graphs. Recently, Grzesik, Klimošová, Pilipčuk, and Pilipčuk [19] extended this result to $P_6$-free graphs. We also note that if $P_2$-free graphs are polynomial-time solvable on $H$-free graphs for some graph $H$, then it is also polynomial-time solvable on $(P_1 + H)$-free graphs. This follows from a well-known observation for Independent Set (see, for instance, [30]). We can guess a vertex to be in the independent set, remove it and its neighbours and then solve Independent Set in the remaining graph, which is $H$-free.

Theorem 1 ([19]). For every $s \geq 0$, Vertex Cover can be solved in polynomial time for $(sP_1 + P_5)$-free graphs.

By using the concept of the price of connectivity [7,9,23], Chiarelli et al. [10] proved that Connected Vertex Cover is polynomial-time solvable for $sP_5$-free graphs for any integer $s \geq 1$. For Vertex Cover this follows from combining two classical results [3,34] (as is well known). No other complexity results are known for Connected Vertex Cover for $H$-free graphs if $H$ is a linear forest.

Our Result. We continue the study of [10,13] and prove the following result.

Theorem 2. For every $s \geq 0$, Connected Vertex Cover can be solved in polynomial time for $(sP_1 + P_5)$-free graphs.

Our method is based on an analysis of the structure of dominating sets in $(sP_1 + P_5)$-free graphs using a characterization of $P_5$-free graphs due to Bacsó and Tuza [1]. This enables us to translate the problem into a problem in which we try to extend a partial vertex cover into a full connected vertex cover. We solve this extension variant of Connected Vertex Cover by using Theorem 1 (applied to the smaller class of $(sP_1 + P_5)$-free graphs). We show how to do this in Section 3 and then show how to use this result to prove Theorem 2 in Section 4.

We note that for Connected Vertex Cover one cannot extend results on $H$-free graphs to results on $(sP_1 + H)$-free graphs in a straightforward way (certainly one cannot use the technique for Vertex Cover described before Theorem 1). Hence a non-trivial amount of additional work is required to deal with the $s$ forbidden isolated vertices in Theorem 2. Generally speaking, there exist problems whose complexity
jumps from polynomial-time solvable on $P_r$-free graphs to NP-complete on $(P_1 + P_r)$-free graphs. To give an example, COLOURING is polynomial-time solvable for $P_2$-free graphs but is NP-complete for $(P_1 + P_4)$-free graphs [27]. To give another example, a clique transversal of a graph $G$ is a set $S \subseteq V_G$ such that $S$ contains a vertex of each maximal clique of $G$ (note that a vertex cover can be viewed as a transversal which contains a vertex of each 2-vertex clique). It is known that computing a smallest clique transversal can be done in polynomial time for comparability graphs [2] and thus for $P_4$-free graphs, but is NP-hard for cobipartite graphs [20] and thus for $(P_1 + P_4)$-free graphs. Due to these examples, Theorem 2 is somewhat unexpected.

The class of $P_5$-free graphs has also been studied for other problems than VERTEX COVER and CONNECTED VERTEX COVER. Hoàng et al. [25] proved that $k$-COLOURING is polynomial-time solvable for $P_5$-free graphs for every $k \geq 1$. This result was possible to extend to $(sP_1 + P_5)$-free graphs for any $s \geq 0$, as shown by Couturier et al. [11]. Golovach and Heggernes [18] gave a fixed-parameter tractable algorithm for CHOOSABILITY on $P_5$-free graphs when parameterized by the size of the lists of admissible colours. Recently, Bonamy et al. [9] proved that the problems INDEPENDENT FEEDBACK VERTEX SET and INDEPENDENT ODD CYCLE TRANSVERSAL are polynomial-time solvable for $P_5$-free graphs.

**Relation to Contractibility Problems.** A connected graph $G$ on $n$ vertices has a connected vertex cover of size $k$ if and only if $G$ contains the star $K_{1,n-k}$ on $n-k+1$ vertices as a contraction. Hence the optimization version of CONNECTED VERTEX COVER is equivalent to the problem of computing the size of a largest star $K_{1,k}$ to which a connected graph $G$ can be contracted.

Due to the above relationship, CONNECTED VERTEX COVER is related to a number of similar contractibility problems involving elementary graph classes. For instance, the cyclicity of a graph is the size of a largest cycle to which a given graph can be contracted. Cyclicity was introduced by Blum [4] under the name co-circularity. Hammack [21] introduced its current name and proved that the problem of determining the cyclicity is NP-hard [22]. Later it was shown that the problem stays NP-hard for claw-free graphs [15] and bipartite graphs [12].

To give another example, the contraction analogue of LONGEST PATH and LONGEST INDUCED PATH is called LONGEST PATH CONTRACTIBILITY [26]. This problem is to determine the length of a longest path to which a graph can be contracted. It is known to be NP-hard [3] even for line graphs [15], bipartite graphs [21] and $P_5$-free graphs [26], but polynomial-time solvable for chordal graphs [24] and $P_5$-free graphs [26] (in particular note the complexity jump for $P_r$-free graphs from $r = 5$ to $r = 6$).

## 2 Preliminaries

We consider only finite, undirected graphs without multiple edges or self-loops. Let $G = (V, E)$ be a graph. For a set $S \subseteq V$, the graph $G[S]$ denotes the subgraph of $G$ induced by $S$, and we say that $S$ is connected if $G[S]$ is connected. We write $G - S = G[V \setminus S]$, and if $S = \{u\}$ we may simply write $G - u$. For a vertex $u \in V$, we write $N_G(u) = \{v \mid uv \in E\}$ to denote the neighbourhood of $u$. For a set $S \subseteq V$,

---

1 If $G$ has a connected vertex cover $S$ of size $k$, then contracting every edge between vertices in $S$ modifies $G$ into $K_{1,n-k}$. If $G$ contains $K_{1,n-k}$ as a contraction, then $V_G$ can be partitioned into sets $A$, $B_1$, . . . , $B_{n-k}$ that each induce a connected graph such that there exists at least one edge between a vertex from $A$ and a vertex from $B_i$ for $i = 1, \ldots, n-k$ and no edges between two vertices from different $B$-sets. If $|B_i| \geq 2$, then we move every vertex that is adjacent to a vertex of $A$ to $A$ until we have only one vertex in $B_i$ left. This gives us a connected vertex cover of size $k$. 

3
we write $N_G(S) = (\bigcup_{u \in S} N_G(u)) \setminus S$. A subset $D \subseteq V$ is a dominating set of $G$ if every vertex of $V \setminus D$ is adjacent to at least one vertex of $D$. An edge $uv$ of a graph $G = (V, E)$ is dominating if $\{u, v\}$ is dominating. The contraction of an edge $uv \in E$ is the operation that replaces $u$ and $v$ by a new vertex that is made adjacent to precisely those vertices that were adjacent to $u$ or $v$ in $G$ (without introducing self-loops or multiple edges).

Let $H$ be a graph. Then a graph $G$ is $H$-free if it does not contain an induced subgraph isomorphic to $H$. The disjoint union $G + H$ of two vertex-disjoint graphs $G$ and $H$ is the graph $(V_G \cup V_H, E_G \cup E_H)$. The disjoint union of $r$ copies of a graph $G$ is denoted by $rG$. A linear forest is the disjoint union of one or more paths.

The graph $P_r$ denotes the path on $r$ vertices. We will use the following result due to Bacsó and Tuza [1] as a lemma in our proof. It is not difficult to compute the set $D$ in polynomial time via a step-by-step vertex approach; this also follows from a more general result of Camby and Schaudt [8] for $P_r$-free graphs ($r \geq 1$).

**Lemma 1 ([1]).** Every connected $P_5$-free graph $G$ has a dominating set $D$, computable in $O(n^3)$ time, that either induces a $P_3$ or a complete graph.

### 3 An Auxiliary Problem

In this section we prove that a variant of CONNECTED VERTEX COVER can be solved in polynomial time for $(sP_1 + P_5)$-free graphs for every integer $s \geq 0$. We start with the following lemma.

**Lemma 2.** Let $s \geq 0$ and let $G$ be a connected $(sP_1 + P_5)$-free graph. Then $G$ has a connected dominating set $D$ that is either a clique or has size at most $2s^2 + s + 2$. Moreover, $D$ can be found in $O(n^{2s^2 + s + 4})$ time.

**Proof.** If $G$ is $P_5$-free, then we apply Lemma [1]. Otherwise, as $G$ is $(sP_1 + P_5)$-free, there exists an integer $0 \leq r \leq s - 1$ such that $G$ contains an induced subgraph $H$ isomorphic to $rP_1 + P_5$. Let $V_H = \{a_1, \ldots, a_r, b_1, \ldots, b_5\}$ such that the $b$-vertices induce a $P_5$ in that order. We choose $r$ to be maximum so $G$ contains no induced $(r + 1)P_1 + P_5$. Hence, $V_H$ dominates $G$. As $G$ is $(sP_1 + P_5)$-free, $G$ is $P_{5+2s}$-free. Hence, for each $a_i$, there exists a path of at most $5 + 2s - 1$ vertices that connects $a_i$ to $b_1$. Let $H^*$ be the graph that contains $H$ and all these $a_i - b_1$-paths. Then we choose $D = V_H \cup H^*$. As $V_H$ dominates $G$, we find that $D \supseteq V_H$ also dominates $G$. Moreover, $D$ has size at most $r(5 + 2s - 2) + 5 \leq 2s^2 + s + 2$ vertices. We can find $D$ by considering, if needed, every set of at most $2s^2 + s + 2$ vertices in $G$ and checking if such a set is dominating. This brute force procedure takes $O(n^{2s^2 + s + 4})$ time in total. \hfill $\square$

It is readily seen that edge contractions do not destroy connectivity or $(sP_1 + P_5)$-freeness.

**Lemma 3.** Let $G$ be a connected $(sP_1 + P_5)$-free graph for some $s \geq 0$. The graph obtained from $G$ after contracting an edge is also connected and $(sP_1 + P_5)$-free.

To prove Theorem 2 we will solve a polynomial number of instances of a variant of CONNECTED VERTEX COVER, which we show can be solved in polynomial time for $(sP_1 + P_5)$-free graphs for every $s \geq 0$. This variant is defined as follows. We first describe its input. Let $G$ be a connected graph, let $J \subseteq V_G$ be a subset of the vertex set of $G$ and let $y$ be a vertex of $J$. We call the triple $(G, J, y)$ cover-complete if it has the following properties:

(A) $J$ is an independent set;
(B) $y$ is adjacent to every vertex of $G - J$;
(C) the neighbours of each vertex in $J \setminus \{y\}$ form an independent set in $G - J$.

We now describe the problem.

**Connected Vertex Cover Completion**

**Instance:** a cover-complete triple $(G, J, y)$.
**Goal:** find a smallest connected vertex cover $S$ of $G$ such that $J \subseteq S$.

We will show how to solve Connected Vertex Cover Completion in polynomial time for $(sP_1 + P_3)$-free graphs for any $s \geq 0$. In order to do this we first prove a number of lemmas.

Let $(G, J, y)$ be a cover-complete triple, where $G$ is a connected $(sP_1 + P_3)$-free graph. For a vertex $w \in N_G(J \setminus \{y\})$, we write $J_w = N_G(w) \cap J$. Note that $y \in J_w$ due to (B). Now let $G'$ be the graph obtained from $G$ after contracting every edge of $G[J_w \cup \{w\}]$. As $G[J_w \cup \{w\}]$ is connected, contracting every edge in $G[J_w \cup \{w\}]$ means that the vertices of $J_w \cup \{w\}$ are merged into a single vertex which we denote by $y_w$. We say that we have set-contracted $G$ into $G'$ via $w$ and that we contracted $J_w \cup \{w\}$ into $y_w$. We will now prove a crucial lemma.

**Lemma 4.** Let $(G, J, y)$ be a cover-complete triple, where $G$ is a connected $(sP_1 + P_3)$-free graph for some $s \geq 0$. Let $w \in N_G(J \setminus \{y\})$, and let $G'$ be the graph obtained from $G$ after set-contracting via $w$. Let $J' = (J \setminus J_w) \cup \{y_w\}$ and $y' = y_w$. Then the following holds:

1. $G'$ is a connected $(sP_1 + P_3)$-free graph;
2. $(G', J', y')$ is a cover-complete triple;
3. A set $S \subseteq V_G$ is a (smallest) connected vertex cover of $G$ that contains $J \cup \{w\}$ if and only if $(S \setminus (J \cup \{w\}) \cup J'$ is a (smallest) connected vertex cover of $G'$ that contains $J'$.

**Proof.** We will prove 1-3 separately.

1. By Lemma 3 $G'$ is connected and $(sP_1 + P_3)$-free. This proves 1.

2. We will prove (A)-(C) for $(G', J', y')$. Before we do this we first observe the following. As (B) holds for $(G, J, y)$, we find that $y \in J$ is adjacent to $w$ in $G$. Hence $y$ belongs to $J_w$ and thus to $J_w \cup \{w\}$, which is contracted to the single vertex $y'$ in $G'$. Hence, $y$ is not in $G'$ and its role has been taken over by $y'$, as we show below.

   We first prove (A). As $J$ is an independent set in $G$, we find that $J \setminus J_w$ is an independent set in $G'$. For contradiction, suppose that $y'$ is adjacent to a vertex in $J \setminus J_w$. Then there is an edge between a vertex of $J \setminus J_w$ and a vertex of $J_w \cup \{w\}$ in $G$. However, this is not possible as $J$ is independent in $G$, and thus every edge in $G[J \cup \{w\}]$ is incident with $w$. Hence $J' = (J \setminus J_w) \cup \{y'\}$ is an independent set in $G'$. This proves (A).

   We now prove (B). Recall that $y$ belongs to $J_w \cup \{w\}$, which is contracted to $y'$ in $G'$. Hence, as $y$ is adjacent to every vertex of $G - J$ in $G$, we find that $y'$ is adjacent to every vertex of $G' - J'$. This proves (B).

   Finally we prove (C). Let $x \in J' \setminus \{y'\}$. Then $x$ is not adjacent to $y'$, as we showed above that $J'$ is an independent set in $G'$. Then $N_{G'}(x) = N_G(x)$ is an independent set, as (C) holds for $(G, J, y)$. This proves (C).

3. Any connected vertex cover $S$ of $G$ that contains $J \cup \{w\}$ contains every vertex of $J_w \cup \{w\}$. Hence contracting $J_w \cup \{w\}$ to $y'$ yields a connected vertex cover $(S \setminus (J \cup \{w\})) \cup J'$ of $G'$ that contains $J'$ Any connected vertex cover $S'$ of $G'$ that contains $J'$ contains $y'$. Hence uncontracting the edges of $G[J_w \cup \{w\}]$ yields a connected vertex cover $(S' \cup J \cup \{w\}) \setminus J'$ of $G$ that contains $J \cup \{w\}$. This proves 3. \qed
Let \((G, J, y)\) be a cover-complete triple. We define \(L_J = N_G(J \setminus \{y\})\). If there is no ambiguity, we will just write \(L = L_J\). Note that \(L\) is the union of a number of independent sets due to (C), but \(L\) itself may not be independent. However we can deduce the following lemma, which follows from (C).

**Lemma 5.** Let \((G, J, y)\) be a cover-complete triple. If \(w_1\) and \(w_2\) are two adjacent vertices in \(L\), then no vertex of \(J \setminus \{y\}\) is adjacent to both \(w_1\) and \(w_2\).

We now introduce some important notions for a cover-complete triple \((G, J, y)\). Two vertices \(w_1, w_2 \in L\) form a pseudo-dominating pair if

- \(w_1\) and \(w_2\) are non-adjacent;
- \(w_1\) has a neighbour in \(J\) not adjacent to \(w_2\); and
- \(w_2\) has a neighbour in \(J\) not adjacent to \(w_1\).

Three vertices \(w_1, w_2, w_3 \in L\) form a pseudo-dominating triple if

- \(w_1\) is adjacent to neither \(w_2\) nor \(w_3\);
- \(w_2\) and \(w_3\) are adjacent;
- \(J\) contains two distinct vertices \(x_1\) and \(x_2\) such that
  - \(x_1 \in N_G(w_1) \setminus N_G(\{w_2, w_3\})\)
  - \(x_2 \in (N_G(w_1) \cap N_G(w_2)) \setminus N_G(w_3)\).

Let \(S\) be a connected vertex cover of \(G\) that contains \(J\). A subset \(L^* \subseteq L \cap S\) is a connector of \(S\) if \(J \cup L^*\) is connected. We prove the following two lemmas.

**Lemma 6.** Let \((G, J, y)\) be a cover-complete triple, where \(G\) is an \((sP_1 + P_5)\)-free graph for some \(s \geq 0\). Let \(S\) be a connected vertex cover of \(G\) that contains \(J\). If \(S\) contains both vertices of a pseudo-dominating pair \(w_1, w_2\), then \(S\) has a connector of size at most \(s + 1\) that contains both \(w_1\) and \(w_2\).

**Proof.** By definition, there exist two vertices \(x_1\) and \(x_2\) in \(J\), such that \(w_1\) is not adjacent to \(x_2\) and \(w_2\) is not adjacent to \(x_1\). As \(J\) is an independent set by (A) and each vertex of \(L\) is adjacent to \(y\) by (B), we find that \(\{x_1, w_1, y, w_2, x_2\}\) induces a \(P_5\) in that order. As \(G\) is \((sP_1 + P_5)\)-free and \(J\) is an independent set, this means that \(\{w_1, w_2\}\) dominates all vertices of \(J\) except for a subset \(I \subseteq J\) of at most \(s - 1\) vertices. We choose \(L^*\) to consist of \(w_1, w_2\) and a neighbour in \(L \cap S\) of each vertex of \(I\) (note that such a neighbour must exist for each vertex of \(I\) as \(S\) is connected). Then \(J \cup L^*\) is connected, that is, \(L^*\) is a connector, as each vertex of \(J\) is adjacent to some vertex of \(L^*\) and each vertex of \(L^*\) is adjacent to \(y \in J\) due to (B). Moreover, \(L^*\) has size at most \(s + 1\).

**Lemma 7.** Let \((G, J, y)\) be a cover-complete triple, where \(G\) is an \((sP_1 + P_5)\)-free graph for some \(s \geq 0\). Let \(S\) be a connected vertex cover of \(G\) that contains \(J\). If \(S\) contains all three vertices of a pseudo-dominating triple \(w_1, w_2, w_3\), then \(S\) has a connector of size at most \(s + 2\) that contains \(\{w_1, w_2, w_3\}\).

**Proof.** By definition, there exist two vertices \(x_1\) and \(x_2\) in \(J\) such that \(x_1\) is adjacent to \(w_1\) but not to \(w_2\) and \(w_3\), and \(x_2\) is adjacent to \(w_1\) and \(w_2\) but not \(w_3\). Then \(\{x_1, w_1, x_2, w_2, w_3\}\) induce a \(P_5\) in that order. As \(G\) is \((sP_1 + P_5)\)-free and \(J\) is an independent set, this means that \(\{w_1, w_2, w_3\}\) dominates all vertices of \(J\) except for a subset \(I \subseteq J\) of at most \(s - 1\) vertices. We choose \(L^*\) to consist of \(w_1, w_2, w_3\) and a neighbour in \(L \cap S\) of each vertex of \(I\) (note that such a neighbour must exist for each vertex of \(I\) as \(S\) is connected). Then \(J \cup L^*\) is connected, that is, \(L^*\) is a connector, as each vertex of \(J\) is adjacent to some vertex of \(L^*\) and each vertex of \(L^*\) is adjacent to \(y \in J\) due to (B). Moreover, \(L^*\) has size at most \(s + 2\).
Let \((G, J, y)\) be a cover-complete triple. Let \(S\) be a connected vertex cover of \(G\) that contains \(J\). If \(S\) contains both vertices of some pseudo-dominating pair of \(G\) or all three vertices of some pseudo-dominating pair of \(G\), then \(S\) is of type 1. Otherwise \(S\) must contain at most one vertex of any pseudo-dominating pair and at most two vertices of any pseudo-dominating triple of \(G\). In that case we say that \(S\) is of type 2.

We observe that \(G\) might have connected covers of only one type.

We will now prove how to find a smallest type-1 connected vertex cover of a graph \(G\) of a cover-complete triple \((G, J, y)\) in polynomial time (if it exists). Afterwards we prove how to find a smallest type-2 connected vertex cover of \(G\) in polynomial time (if it exists). In order to compute these sets we need the following lemma, which also explains how we will use Theorem 1 in our proof.

**Lemma 8.** Let \((G, \{y\}, y)\) be a cover-complete triple, where \(G\) is an \((sP_1 + P_5)\)-free graph for some \(s \geq 0\). Then it is possible to compute a smallest connected vertex cover of \(G\) that contains \(y\) in polynomial time.

**Proof.** As \((G, \{y\}, y)\) is a cover-complete triple, \(y\) dominates \(G\). Moreover \(G - y\) is \((sP_1 + P_5)\)-free. Then we can compute a smallest vertex cover \(S\) of \(G - y\) in polynomial time by Theorem 1. As \(y\) dominates \(G\), we find that \(S \cup \{y\}\) is a smallest connected vertex cover of \(G\) that contains \(y\).

We are now ready to deal with type-1 smallest connected vertex covers.

**Lemma 9.** Let \((G, J, y)\) be a cover-complete triple. Then it is possible to find in polynomial time a smallest type-1 connected vertex cover of \(G\).

**Proof.** We find pseudo-dominating pairs of \(G\) by brute force. Note that this takes polynomial time. For each pseudo-dominating pair \((w_1, w_2)\) of \(G\), we describe how to compute a smallest connected vertex cover \(S_{w_1, w_2}\) of \(G\) that contains \(J \cup \{w_1, w_2\}\). By Lemma 8 such a vertex cover must have a connector \(L^*\) of size at most \(s + 1\) that contains \(w_1\) and \(w_2\). We guess \(L^* \setminus \{w_1, w_2\}\). As there are \(O(n^{s-1})\) sets to guess, this adds only a polynomial factor to the running time of our algorithm. For each guess \(L^*\), we check if \(J \cup L^*\) is connected. If so, then we apply Lemma 8 recursively for each \(w \in L^*\). As \(|L^*| \leq s + 1\), this takes polynomial time. Let \((G', J', y')\) be the resulting cover-complete triple. Then \(J' = \{y'\}\), which means we can apply Lemma 8 to find a smallest connected vertex cover \(S'\) of \(G'\) in polynomial time. We translate \(S'\), in polynomial time, into the desired vertex cover \(S_{w_1, w_2}\) by uncontracting any contracted edges using Lemma 8.

For each pseudo-dominating triple \((w_1, w_2, w_3)\) of \(G\) we compute a smallest connected vertex cover \(S_{w_1, w_2, w_3}\) of \(G\) that contains \(J \cup \{w_1, w_2, w_3\}\). We can do this in polynomial time by exactly the same arguments: the only difference is that we need to apply Lemma 8 and guess a connector \(L^*\) of size at most \(s + 2\).

From all the computed sets \(S_{w_1, w_2}\) and \(S_{w_1, w_2, w_3}\) we take the smallest one (which we can do in polynomial time). This proves Lemma 9.

Let \((G, J, y)\) be a cover-complete triple. Due to Lemma 9 we can deal with the connected vertex covers of \(G\) that are of type 1. However, it might be possible that \(G\) has a smaller connected vertex cover of type 2. In order to find out about this, we will introduce the following two reduction rules that will transform a cover-complete triple \((G, J, y)\) into a triple \((G', J', y')\) with \(|J'| < |J|\). We say that such a rule is safe if the following three statements hold:

1. If \(G\) is \((sP_1 + P_5)\)-free and connected, then \(G'\) is \((sP_1 + P_5)\)-free and connected.
2. \((G', J', y')\) is cover-complete.
3. Given a smallest connected vertex cover \( S' \) of \( G' \) that contains \( J' \), it is possible, in polynomial time, to find a smallest connected vertex cover \( S \) of \( G \) that contains \( J \).

**Rule 1.** Set-contract via \( x \) whenever \( x \) is a vertex in \( L \cap N_G(w_1) \cap N_G(w_2) \) for some pseudo-dominating pair \((w_1, w_2)\).

**Lemma 10.** Rule 1 is safe.

**Proof.** Let \((G', J', y')\) be the resulting triple after an application of Rule 1, where \( J' = (J \setminus J_x) \cup \{y_x\} \) and \( y' = y_x \). By Lemma 4 \((G', J', y')\) is a cover-complete triple. By the same lemma, \( G' \) is \((sP_1 + P_5)\)-free and connected if \( G \) is \((sP_1 + P_5)\)-free and connected. Hence we have proven that conditions 1 and 2 hold.

We are left to prove condition 3. Let \( S' \) be a smallest connected vertex cover in \( G' \) that contains \( J' \). Then \( S = (S' \setminus \{y'\}) \cup J_x \cup \{x\} \) is a smallest connected vertex cover of \( G \) that contains \( J \cup \{x\} \) due to Lemma 11. We prove the following claim.

**Claim.** For any type-2 connected vertex cover \( T \) of \( G \), it holds that \(|T| \geq |S|\).

We prove the Claim as follows. Let \( T \) be a connected vertex cover \( T \) of \( G \) that is of type 2. Suppose \( x \notin T \). Then, as \( x \) is adjacent to both \( w_1 \) and \( w_2 \), we find that \( T \) contains both \( w_1 \) and \( w_2 \). Thus \( T \) is not of type 2, a contradiction. Hence \( T \) contains \( x \). This implies that the set \( T' = (T \setminus \{x\}) \cup J_x \) is a connected vertex cover of \( G' \) that contains \( J' \). As \( S' \) is a smallest connected vertex cover of \( G' \) that contains \( J' \), we find that \(|T'| \geq |S'|\). Hence \(|T| = |T'| + |J_x| \geq |S'| + |J_x| = |S|\). This proves the Claim.

The above means that we can do as follows. Given \( S' \) we compute \( S \) in polynomial time.

By Lemma 9 we can also compute, in polynomial time, a smallest type-1 connected vertex cover \( S^* \) of \( G \) (note that \( S = S^* \) is possible). If \( S \) is of type 2, then \( S \) is a smallest type-2 connected cover of \( G \), due to the Claim. We compare \(|S| \) and \(|S^*| \) and choose the smallest one. If \( S \) is of type 1, then \( S^* \) is a smallest connected vertex cover of \( G \), again due to the Claim. This proves condition 3.

**Rule 2.** For any vertex \( w_5 \in L \) that is not adjacent to any vertex of a clique of four vertices \( w_1, w_2, w_3, w_4 \) in \( L \), delete \( w_5 \) and set-contract via \( u \) for every \( u \in L \cap N_G(w_5) \).

**Lemma 11.** Rule 2 is safe.

**Proof.** We first show that \( w_5 \) cannot be in any connected vertex cover \( S \) of \( G \) that is of type 2. For contradiction, suppose that \( w_5 \) is in such a connected cover \( S \). Because \( S \) is a vertex cover and \( \{w_1, w_2, w_3, w_4\} \) is a clique, \( S \) contains at least three of \( \{w_1, w_2, w_3, w_4\} \), say \( w_1, w_2, w_3 \).

For \( i = 1, \ldots, 5 \), let \( X_i \) be the set of neighbours of \( w_i \) in \( J \). As \( w_i \in L \), every \( X_i \neq \emptyset \) by definition of \( L \). By Lemma 5 we find that \( X_1 \cap X_2 \cap X_3 = \emptyset \). Let \( x \in X_1 \). If \( x \notin X_5 \), then \( X_5 \subseteq X_1 \), otherwise \((w_1, w_5)\) is a pseudo-dominating pair of vertices that are both contained in \( S \), which is not possible as \( S \) is of type 2. As \( X_1 \cap X_2 = \emptyset \), we find that \( X_5 \cap X_2 = \emptyset \). This means that \((w_2, w_5)\) is a pseudo-dominating pair of vertices that are both contained in \( S \), which is not possible either. Hence \( x \in X_5 \). We conclude that \( X_1 \subseteq X_5 \). For the same reason, we find that \( X_2 \subseteq X_5 \) and \( X_3 \subseteq X_5 \).

Recall that \( X_1 \cap X_2 \cap X_3 = \emptyset \). Hence we can pick a vertex \( x_1 \in X_1 \) and a vertex \( x_3 \in X_3 \), which are both adjacent to \( w_5 \) but not to \( w_2 \), in order to find that \((w_5, w_1, w_2)\) is a pseudo-dominating triple. As all three vertices \( w_1, w_2, w_5 \) belong to \( S \), while \( S \) is of type 2, this is not possible. Hence \( S \) does not contain \( w_5 \).

As no connected vertex cover of \( G \) of type 2 may contain \( w_5 \), any connected vertex cover of \( G \) that is of type 2 must contain all neighbours of \( w_5 \), and we can delete \( w_5 \). The proof of conditions 1–3 is identical to the proof of Lemma 10 where the neighbours of \( w_5 \) in \( L \) take the role of the vertex \( x \) in the proof of Lemma 10. 

\[ \square \]
We call a cover-complete triple \((G, J, y)\) free if \(G\) has no pseudo-dominating pair with a common neighbour in \(L\), and moreover, \(G[L]\) is \((P_1 + K_4)\)-free. We prove the following lemma.

**Lemma 12.** It is possible to modify, in polynomial time, a cover-complete triple \((G, J, y)\) into a free cover-complete triple \((G', J', y)\) with the following properties:

1. If \(G\) is \((sP_1 + P_5)\)-free and connected, then \(G'\) is \((sP_1 + P_5)\)-free and connected.
2. Given a smallest connected vertex cover \(S'\) of \(G'\) that contains \(J'\), it is possible to find in polynomial time a smallest connected vertex cover \(S\) of \(G\) that contains \(J\).

**Proof.** We first apply Rule 1 exhaustively and afterwards we apply Rule 2 exhaustively. As each application of each of these rules takes polynomial time and reduces \(|V_G|\), this procedure will stop after polynomial time. By repeated use of Lemmas 10 and 11, this results in a cover-complete triple \((G', J', y)\) that satisfies the two properties of the lemma. Moreover, \(G'\) contains no pseudo-dominating pair with a common neighbour in \(L' = L'_{+1}\) and \(G'[L']\) is \((P_1 + K_4)\)-free, as otherwise we could still apply Rule 1 or Rule 2, respectively. \(\square\)

Let \((G, J, y)\) be a free cover-complete triple. A connector of a connected vertex cover \(S\) of \(G\) is minimal if it does not properly contain a smaller connector of \(S\). We will prove two structural and one algorithmic lemma on free cover-complete triples.

**Lemma 13.** Let \((G, J, y)\) be a free cover-complete triple. Every minimal connector \(L^*\) of every type-2 connected vertex cover \(S\) of \(G\) is a clique.

**Proof.** For contradiction, suppose that \(L^*\) is not a clique. Then \(L^*\) contains two non-adjacent vertices \(w_1\) and \(w_2\). As \(L^*\) is a minimal connector, \(w_1\) has a neighbour in \(J\) not adjacent to \(w_2\), and vice versa. However, then \((w_1, w_2)\) is a pseudo-dominating pair of \(G\). This is not possible, as \(S\) is of type 2. \(\square\)

**Lemma 14.** Let \((G, J, y)\) be a free cover-complete triple that has a pseudo-dominating pair \((w_1, w_2)\). Then every minimal connector \(L^*\) of every type-2 connected vertex cover \(S\) of \(G\) has size at most 5.

**Proof.** For contraction, suppose that \(|L^*| \geq 6\). By Lemma 13, \(L^*\) is a clique. As \((G, J, y)\) is free, \(G'[L']\) is \((K_4 + P_1)\)-free by definition. Hence \(w_1\) must be adjacent to at least three vertices of \(L^*\), which we denote by \(x_1, x_2, x_3\). Note that \(\{w_1, x_1, x_2, x_3\}\) induces a \(K_4\) in \(G[L]\). By definition of a pseudo-dominating pair, \(w_1\) and \(w_2\) are non-adjacent. As \((G, J, y)\) is free, \(w_2\) is not adjacent to any neighbour of \(w_1\) in \(L\) by definition. Hence \(w_2\) is not adjacent to any vertex of \(\{x_1, x_2, x_3\}\). This means that the set \(\{w_1, w_2, x_1, x_2, x_3\}\) induces a \(K_4 + P_1\) in \(G[L]\), a contradiction. \(\square\)

**Lemma 15.** Let \((G, J, y)\) be a free cover-complete triple that has no pseudo-dominating pair. It is possible to find in polynomial time a clique \(K \subseteq L\) with \(N_G(K) \cap J = J\).

**Proof.** We describe how to construct \(K\). Consider a vertex \(w_1 \in L\) that has maximum neighbourhood in \(J\), that is, there is no vertex \(w \in L\) with \(N_G(w_1) \cap J \subseteq N_G(w) \cap J\). We put \(w_1\) in \(K\). Suppose that at some point we have constructed a clique \(K = \{w_1, \ldots, w_i\}\) for some \(i \geq 1\). If \(N_G(K) \cap J = J\), then we stop. Otherwise we pick a vertex \(w_{i+1}\) with maximum neighbourhood in \(J \setminus N_G(K)\) over all vertices in \(L\) (or equivalently, all vertices in \(L \setminus \{w_1, \ldots, w_i\}\)). Note that \(w_{i+1}\) exists as \(G\) is connected. Suppose that \(w_{i+1}\) is adjacent to some \(x \in N_G(K) \cap J\). Then, by Lemma 5, we find that \(x\) is adjacent to a unique vertex \(w_h\) in \(K\). By the same lemma, \(w_{i+1}\) is not adjacent to \(w_h\). As \(G\) has no pseudo-dominating pair and \(w_{i+1}\) has a neighbour in \(J \setminus N_G(K)\). \(\square\)
(that is, a neighbour not adjacent to \(w_{h}\)), we find that \(N_G(w_h) \subseteq N_G(w_{i+1})\). This means that we would have chosen \(w_{i+1}\) earlier, namely instead of \(w_h\). Hence, \(w_{i+1}\) is not adjacent to any \(x \in N_G(K) \cap J\). As \(G\) has no pseudo-dominating pairs, this means that \(w_{i+1}\) is adjacent to every \(w_j\) with \(1 \leq j \leq i\). That is, we can extend \(K\) into a larger clique by adding \(w_{i+1}\).

As we increase \(N_G(K) \cap J\) each time we add a new vertex to \(K\), our procedure will stop with the desired output \(K = \{w_1, \ldots, w_r\}\) for some \(r \geq 1\). We note that constructing \(K\) takes polynomial time. \(\square\)

We are now ready to prove the following theorem.

**Theorem 3.** For every \(s \geq 0\), CONNECTED VERTEX COVER COMPLETION can be solved in polynomial time for \((sP_1 + P_5)\)-free graphs.

**Proof.** Let \(s \geq 0\) and let \((G, J, y)\) be a cover-complete triple, where \(G\) is an \((sP_1 + P_5)\)-free graph. We first apply Lemma 12 to obtain a free cover-complete triple \((G', J', y')\) in polynomial time. By the same lemma, \(G'\) is \((sP_1 + P_5)\)-free. Our aim is to find a smallest connected vertex cover of \(G'\) that contains \(J'\) in polynomial time, so that we can apply statement 2 of Lemma 12. We first compute in polynomial time a smallest type-1 connected vertex cover \(S'\) of \(G'\) by using Lemma 9. We now need to compute a smallest type-2 connected vertex cover \(S'\) of \(G'\) and compare \(|S'|\) with \(|S'|\).

First suppose that \(G'\) contains a pseudo-dominating pair. We guess a minimal connector of size at most 5 and apply Lemma 4 on its vertices. If we obtain an instance \((G'', \{y''\}, y'')\), then we apply Lemma 8. Afterwards we uncontract all contracted edges to get a connected vertex cover of \(G''\) of type 2. By Lemma 14 doing this for every guessed minimal connector of size at most 5 gives us a smallest type-2 connected vertex cover \(S'\) of \(G''\). As we process each guess in polynomial time and there are at most \(O(n^5)\) guesses, we found \(S'\) in polynomial time. We compare \(|S'|\) and \(|S'|\) and choose the smallest one.

Now suppose that \(G''\) has no pseudo-dominating pair. Let \(L' = N_{G'}(J' \setminus \{y'\})\). By Lemma 15 we obtain in polynomial time a clique \(K \subseteq L'\) with \(N_{G''}(K) \cap J' = J'\). Let \(K = \{w_1, \ldots, w_r\}\) for some \(r \geq 1\). As \(K\) is a clique, every vertex cover contains at least \(r - 1\) vertices of \(K\). This means we will do as follows. We are first going to find in polynomial time a smallest connected vertex cover of \(G''\) that contains \(K \cup J'\). Afterwards, we will find in polynomial time, for \(i = 1, \ldots, r\), for a smallest connected vertex cover of \(G''\) that contains \(J' \cup (K \setminus \{w_i\})\) and that does not contain \(w_i\). As there are \(O(n)\) cases, the total time is polynomial.

We start with computing a smallest connected vertex cover of \(G''\) that contains \(K \cup J'\). We do so by set-contracting via each vertex of \(K\). By Lemma 4 this yields a cover-complete triple \((G'', \{y''\}, y'')\) on which we apply Lemma 8. Afterwards we uncontract all contracted edges in polynomial time. By Lemma 4 this yields a smallest connected vertex cover \(S_K\) of \(G''\) that contains \(J' \cup K\).

We will now show how to compute, in polynomial time, a smallest connected vertex cover of \(G''\) that contains \(J' \cup K \setminus \{w_1\}\) and that does not contain \(w_1\). The case \(i \geq 2\) is done in the same way.

Let \(A = L' \setminus N_{G'}(w_1)\) consists of all non-neighbours of \(w_1\) in \(L'\). As \(G'[L']\) is \((K_4 + P_1)\)-free by definition, we find that \(G'[A]\) is \(K_4\)-free. As \(w_1\) is not in the connected vertex cover we are looking for we remove \(w_1\), and we set-contract via each neighbour of \(w_1\) in \(L\). By Lemma 4 we may now consider the resulting cover-complete triple \((G'', J'', y'')\) where \(G''\) is connected and \((sP_1 + P_5)\)-free. We observe that we did not create any new pseudo-dominating pairs. As \(G''\) had no pseudo-dominating pairs, this implies that \(G''\) has no pseudo-dominating pairs. We write \(L'' = N_{G''}(J'' \setminus \{y''\})\). As \(L'' \subseteq A\), we find that \(G''[L'']\) is \(K_4\)-free.

10
Claim. Every minimal connector \( L^* \) of every connected vertex cover of \( G'' \) that contains \( J'' \) has size at most 3.

We prove the Claim by showing that \( L^* \) is a clique, which implies that \( L^* \) has size at most 3, as \( G''[L^*] \) is \( K_4 \)-free. For contradiction, suppose \( L^* \) is not a clique. Then \( L^* \) contains two non-adjacent vertices \( w_1 \) and \( w_2 \). As \( L^* \) is a minimal connector, \( w_1 \) has a neighbour in \( J'' \) not adjacent to \( w_2 \), and vice versa. However, then \((w_1, w_2)\) is a pseudo-dominating pair of \( G'' \). This is not possible, as \( G'' \) has no pseudo-dominating pairs. Hence we have proven the Claim.

We now guess a minimal connector by considering all subsets in \( L'' \) that have size at most 3. For each guess we apply Lemma 4 on its vertices. If we obtain an instance \( J \) has a neighbour in \( G \) then take the smallest one of these connected vertex covers of \( G \). We then take the smallest one of these connected vertex covers of \( G'' \). For this connected vertex cover of \( G'' \) we uncontract all contracted edges again to obtain in polynomial time a smallest connected vertex cover \( S_{w_1} \) of \( G' \) that contains \( J'' \cup (K \setminus \{w_1\}) \) and that does not contain \( w_1 \).

As mentioned, we pick the smallest one out of the connected vertex covers \( S_{w_i} \) for \( 1 \leq i \leq r \) to obtain a smallest type-2 connected vertex cover of \( G' \), the size of which we compare with the size of \( S^* \). We pick the smallest one.

From the above, we obtain in polynomial time a smallest connected vertex cover of \( G' \) that contains \( J' \) (both in the case where \( G' \) has a pseudo-dominating pair and in the case where \( G' \) has no pseudo-dominating pair). As mentioned, it remains to apply statement 2 of Lemma 12 in order to find in polynomial time a smallest connected vertex cover of \( G \) that contains \( J \). The correctness of our algorithm follows immediately from the above case analysis and the description of the cases. \( \Box \)

4 Our Main Result

In this section we prove Theorem 2 that is, we show that CONNECTED VERTEX COVER can be solved in polynomial time for \((sP_1 + P_3)\)-free graphs for every integer \( s \geq 0 \). The proof of our main result heavily relies on Theorem 3. The main idea is to reduce an \((sP_1 + P_3)\)-free input graph \( G \) of CONNECTED VERTEX COVER to a polynomial number of instances \((G_i, J_i, y_i)\) of CONNECTED VERTEX COVER COMPLETION. We then solve each of these instances \((G_i, J_i, y_i)\) of the latter problem in polynomial time by Theorem 3. Afterwards we translate the resulting connected vertex covers of \( G_i \) (which contain \( J_i \)) into connected vertex covers of \( G \). We then pick the smallest one of these sets as our final output. We will also need the following lemma.

Lemma 16. Let \( J \) be an independent set of a connected graph \( G \), such that \( J \) has a vertex \( y \) that is adjacent to every vertex of \( G - J \). Let \( J' \) consist of those vertices of \( J \setminus \{y\} \) that have two adjacent neighbours in \( G - J \) (or equivalently, in \( G \)). Then a subset \( S \) is a connected vertex cover of \( G \) that contains \( J \) if and only if \( S \setminus J' \) is a connected vertex cover of \( G - J' \) that contains \( J \setminus J' \).

Proof. Let \( w \in J \setminus \{y\} \) be a vertex in \( G \) with two adjacent neighbours \( a \) and \( b \) in \( G - J \) (or equivalently in \( G \)). Let \( S \) be a subset of \( G \). First suppose that \( S \) is a connected vertex cover of \( G \) that contains \( J \). Then \( S \setminus \{w\} \) is a vertex cover of \( G - w \) that contains \( J \setminus \{w\} \). As \( y \in J \) and \( y \neq w \), we find that \( S \) contains \( y \). Then every vertex of \( S \) that belongs to \( G - J \) is adjacent to \( y \) in \( G[S] \). Moreover, as \( S \) is connected and \( J \) is independent, every vertex of \( J \setminus \{w\} \) must be adjacent in \( G[S] \) to a vertex of \( G - J \). Hence, \( S \) is connected in \( G - w \).
Now suppose that $S \setminus \{w\}$ is a connected vertex cover of $G - w$ that contains $J \setminus \{w\}$. Then $S$ is a vertex cover of $G$ that contains $J$. As $y \in J$, we find that $S$ contains $y$. As $ab$ is an edge, $S$ contains at least one of $a$ and $b$. Then $w$ and $y$ are adjacent in $S$ either due to the edges $ya$, $aw$ (if $a$ is in $S$) or due to the edges $yb$, $bw$ (if $a$ is not in $S$, as then $b \in S$). Hence $S$ is connected in $G$.

We now consider the graph $G - w$ and repeat the arguments above for any vertex in $J' \setminus \{w\}$.

We are now ready to prove our main result.

**Theorem 2 (Restated)** For every $s \geq 0$, CONNECTED VERTEX COVER can be solved in polynomial time for $(sP_1 + P_5)$-free graphs.

**Proof.** Let $G$ be an $(sP_1 + P_5)$-free graph on $n$ vertices for some $s \geq 0$. We may assume without loss of generality that $G$ is connected. By Lemma 2 we can first compute in $O(n^{2s^2+4})$ time a connected dominating set $D$ that either has size at most $2s^2 + s + 2$ or else is a clique. We note that any vertex cover of $G$ contains all but at most two vertices of $D$ if $D$ is a clique. This leads to a case analysis where we guess the subset $D^* \subseteq D$ of vertices not in a minimum connected vertex cover of $G$. Because $|D^*| \leq 2s^2 + s + 2$, the number of guesses is polynomial. For each guess of $D^*$ we compute a smallest connected vertex cover $S_{D^*}$ that contains all vertices of $D \setminus D^*$ and no vertex of $D^*$. Then in the end we return the one that has minimum size overall.

Let $D^*$ be a guess. Before we start our case analysis we first prove the following claim.

**Claim 1.** We may assume without loss of generality that $D \setminus D^*$ is connected.

We prove Claim 1 as follows. Suppose $D \setminus D^*$ is not connected. Recall that $G[D]$ is either a complete graph or has size at most $2s^2 + s + 2$. In the first case, $G[D \setminus D^*]$ is connected. Hence, the second case applies meaning that $D$ has size at most $2s^2 + s + 2$. Let $v \in D \setminus D^*$. As $G$ is $(sP_1 + P_5)$-free, $G$ is also $P_{2s^2+4}$-free. Hence, for each $u \in D \setminus (D^* \cup \{v\})$, any connected vertex cover of $G$ contains a path of at most $5 + 2s - 1$ vertices that connects $u$ to $v$. We will guess all these $u - v$-paths (using only vertices from $G - D^*$) and add their vertices to $D$. As the number of paths are most $2s^2 + s + 1$, this branching only adds a polynomial factor to our running time and increases our set $D$ by only a constant number of extra vertices. Hence, we have proven Claim 1.

**Case 1.** $D^* = \emptyset$.

We compute a minimum vertex cover $S'$ of $G - D$ in polynomial time by Theorem 1. As $S'$ is a vertex cover of $G - D$, we find that $S' \cup D$ is a vertex cover of $G$. As $D$ is a connected dominating set, $S' \cup D$ is even a connected vertex cover of $G$. Let $S_0 = S' \cup D$. As $S'$ is a minimum vertex cover of $G - D$, $S_0$ is the smallest connected vertex cover of $G$ that contains all vertices of $D$. We remember $S_0$. Note that we obtained $S_0$ in polynomial time.

**Case 2.** $1 \leq |D^*| \leq 2s^2 + s + 2$.

Recall that we are looking for a smallest connected vertex cover of $G$ that contains every vertex of $D \setminus D^*$ and that does not contain any vertex of $D^*$. Hence $D^*$ must be an independent set and $G - D^*$ must be connected (if one of these conditions is false, then we stop considering the guess $D^*$). Moreover, a vertex cover that contains no vertex of $D^*$ must contain all vertices of $N_G(D^*)$. Hence we can safely contract not only any edge between two vertices of $D \setminus D^*$, but also any edge between two vertices in $N_G(D^*)$ or between a vertex of $D \setminus D^*$ and a vertex in $N_G(D^*)$. We perform edge contractions recursively and as long as possible while remembering all the edges that we contract. Let $G^*$ be the resulting graph.
Note that the set \( D^* \) still exists in \( G^* \), as we did not contract any edges with an end-point in \( D^* \). By Claim 1, the set \( D \setminus D^* \) in \( G \) corresponds to exactly one vertex of in \( G^* \). We denote this vertex by \( y \). We observe the following equivalence, which is obtained after uncontracting all the contracted edges.

**Claim 2.** Every smallest connected vertex cover of \( G^* \) that contains \( y \) and that does not contain any vertex of \( D^* \) corresponds to a smallest connected vertex cover of \( G \) that contains \( D \setminus D^* \) and that does not contain any vertex of \( D^* \), and vice versa.

As we obtained \( G^* \) in polynomial time and we can uncontract all contracted edges in polynomial time as well, Claim 2 tells us that we may consider polynomials time as well, Claim 2 tells us that we may consider now consider the graph \( \Lambda \). We translate \( \Lambda \) into a smallest connected vertex cover of the graph \( \Lambda \) in polynomial time and we can uncontract all contracted edges in polynomial time into a smallest connected vertex cover of the graph \( \Lambda \) in polynomial time.

Let \( J \subseteq J^* \) consist of \( y \) and those vertices in \( J^* \) whose neighbourhood in \( G^* \) is an independent set. As \( y \) is adjacent to every vertex of \( (G^* - D^*) - J^* \) in \( G^* - D^* \), and we can remember the set \( J \setminus J^* \), we can apply Lemma 3 and remove \( J \setminus J^* \). That is, it suffices to find a smallest connected vertex cover of the graph \( G' = (G^* - D^*) - (J^* \setminus J) \) that contains \( J \).

As \( J^* \) is an independent set of \( G^* - D^* \), we find that \( J \) is an independent set of \( G' \). By definition, \( y \in J \). As \( y \) is adjacent to every vertex of \( (G^* - D^*) - J^* \) in \( G^* - D^* \), we find that \( y \) is adjacent to every vertex in \( G' - J \). By definition, the neighbours of each vertex in \( J \setminus \{y\} \) form an independent set in \( G' - J \). Hence the triple \( (G', J, y) \) is cover-complete. This means that we can apply Theorem 3 to find in polynomial time a smallest connected vertex cover \( S' \) of \( G' \) that contains \( J \).

We translate \( S' \) in polynomial time into a smallest connected vertex cover \( S^* \) of \( G^* - D^* \) that contains \( J^* \) by adding \( J^* \setminus J \) to \( S' \). We translate \( S^* \) in polynomial time into a smallest connected vertex cover \( S_{D^*} \) of \( G \) that contains no vertex of \( D^* \) by uncontracting any contracted edges.

As mentioned, in the end we pick, in polynomial time, a smallest set of the sets \( S_{D^*} \). This set is then a minimum connected vertex cover of \( G \), which we obtained in polynomial time. We have not sought to optimize the running time of the algorithm so do not provide a detailed analysis, but observe that, for sufficiently large \( s \), it is \( n^{O(s^3)} \). The running time is dominated by obtaining a connected \( D \setminus D^* \) (in Claim 1). As \( D \setminus D^* \) has \( O(n^{2s^2+3s+2}) \) components and the paths required to join them each have \( O(s) \) vertices, the time required to find them is \( n^{O(s^3)} \). The correctness of our algorithm follows immediately from the above case analysis and the description of the cases.

\[ \square \]

5 **Conclusions**

We proved that **Connected Vertex Cover** is polynomial-time solvable for \((sP_1 + P_5)\)-free graphs for every integer \( s \geq 0 \). We finish our paper with posing the following two open problems.

1. What is the complexity of **Connected Vertex Cover** for \( P_6 \)-free graphs?
2. Does there exist an integer $r$ such that CONNECTED VERTEX COVER is NP-complete for $P_r$-free graphs?

For Question 1, it might be easier to consider first the class of $(P_2 + P_3)$-free graphs, for which we do not know the complexity of CONNECTED VERTEX COVER either.

References

1. G. Bacsó and Zs. Tuza, Dominating cliques in $P_7$-free graphs, Periodica Mathematica Hungarica 21(1990) 303–308.
2. V. Balachandhran, P. Nagavamsi, C. Pandu Rangan, Clique transversal and clique independence on comparability graphs, Information Processing Letters 58 (1996) 181–184.
3. E. Balas and C. S. Yu, On graphs with polynomially solvable maximum-weight clique problem, Networks 19 (1989) 247–253.
4. D. Blum, Circularity of graphs, PhD Thesis, Virginia Polytechnic Institute and State University, 1982.
5. M. Bonamy, K.K. Dabrowski, C. Feghali, M. Johnson and D. Paulusma, Independent feedback vertex set for $P_5$-free graphs, Proc. ISAAC 2017, Leibniz International Proceedings in Informatics, to appear.
6. A.E. Brouwer and H.J.Veldman, Contractibility and NP-completeness, Journal of Graph Theory 11 (1987) 71–79.
7. E. Camby, J. Cardinal, S. Fiorini and O. Schaudt, The price of connectivity for vertex cover. Discrete Mathematics & Theoretical Computer Science 16 (2014) 207–224.
8. E. Camby and O. Schaudt, A new characterization of $P_5$-free graphs, Algorithmica 75 (2016), 205–217.
9. J. Cardinal and E. Levy, Connected vertex covers in dense graphs, Theoretical Computer Science 411 (2010) 2581–2590.
10. N. Chiarelli, T.R. Hartinger, M. Johnson, M. Milanic and D. Paulusma, Minimum connected transversals in graphs: new hardness results and tractable cases using the price of connectivity, CoRR, abs/1706.08050, 2017.
11. J.-F. Couturier, P.A. Golovach, D. Kratsch and D. Paulusma, List Coloring in the Absence of a Linear Forest, Algorithmica 71 (2015) 21–35.
12. K.K. Dabrowski and D. Paulusma, Contracting bipartite graphs to paths and cycles, Information Processing Letters 127 (2017) 37–42.
13. B. Escoffier, L. Gourvès and J. Monnot, Complexity and approximation results for the connected vertex cover problem in graphs and hypergraphs, Theoretical Computer Science 8 (2010) 36–49.
14. H. Fernau and D. Manlove, Vertex and edge covers with clustering properties: complexity and algorithms, Journal of Discrete Algorithms 7 (2009) 149–167.
15. J. Fiala, M. Kamiński and D. Paulusma, A note on contracting claw-free graphs, Discrete Mathematics & Theoretical Computer Science 15 (2013) 223–232.
16. M.R. Garey, D.S. Johnson, The rectilinear Steiner tree problem is NP-complete, SIAM Journal on Applied Mathematics 32 (1977) 826–834.
17. F. Gavril, The intersection graphs of subtrees in trees are exactly the chordal graphs Journal of Combinatorial Theory, Series B 16 (1974) 47–56.
18. P. A. Golovach and P. Heggernes, Choosability of $P_5$-free graphs, Proc. MFCS 2009, Lecture Notes in Computer Science 5734 (2009) 382–391.
19. A. Grzesik, T. Klíma, L. Pilipczuk, M. Pilipczuk, and M. Pilipczuk, Polynomial-time algorithm for maximum weight independent set on $P_6$-free graphs, CoRR, abs/1707.05491, 2017.
20. V. Gurugvami and C. Pandu Rangan, Algorithmic aspects of clique-transversal and clique-independent sets, Discrete Applied Mathematics 100 (2000) 183–202.
21. R. Hammack. Cyclicity of graphs, Journal of Graph Theory 32 (1999)160–170.
22. R. Hammack, A note on the complexity of computing cyclicity, Ars Combinatoria 63 (2002) 89–95.
23. T.R. Hartinger, M. Johnson, M. Milanic and D. Paulusma, The price of connectivity for transversals, European Journal of Combinatorics 58 (2016) 203–224.
24. P. Heggernes, P. van ’t Hof, B. Léveque and C. Paul, Contracting chordal graphs and bipartite graphs to paths and trees, Discrete Applied Mathematics 164 (2014) 444–449.

25. C. T. Hoàng, M. Kamiński, V. V. Lozin, J. Sawada, and X. Shu, Deciding $k$-colorability of $P_5$-free graphs in polynomial time, Algorithmica 57 (2010) 74–81.

26. P. van ’t Hof, D. Paulusma and G.J. Woeginger, Partitioning graphs in connected parts, Theoretical Computer Science 410 (2009) 4834–4843.

27. D. Král’, J. Kratochvíl, Zs. Tuza, and G. J. Woeginger, Complexity of coloring graphs without forbidden induced subgraphs, Proc. WG 2001, Lecture Notes in Computer Science 2204 (2001) 254–262.

28. D. Lokshtanov, M. Vatshelle, and Y. Villanger, Independent set in $P_5$-free graphs in polynomial time, Proc. SODA 2014, pages 570–581.

29. G.J. Minty, On maximal independent sets of vertices in claw-free graphs, Journal of Combinatorial Theory, Series B 28 (1980) 307–309.

30. R. Mosca, Stable sets for ($P_6$, $K_2,3$)-free graphs, Discussiones Mathematicae Graph Theory 32 (2012) 387–401.

31. A. Munaro, Boundary classes for graph problems involving non-local properties, Theoretical Computer Science 692 (2017) 46–71.

32. S. Poljak, A note on stable sets and colorings of graphs, Commentationes Mathematicae Universitatis Carolinae. 15 (1974) 307–309.

33. P.K. Priyadarshini and T. Hemalatha, Connected vertex cover in 2-connected planar graph with maximum degree 4 is NP-complete, International Journal of Mathematical, Physical and Engineering Sciences. 2 (2008) 51–54.

34. S. Tsukiyama, M. Ide, H. Ariyoshi, and I. Shirakawa, A new algorithm for generating all the maximal independent sets, SIAM Journal on Computing 6 (1977) 505–517.

35. N. Sbihi, Algorithme de recherche d’un stable de cardinalité maximum dans un graphe sans étoile, Discrete Mathematics 29 (1980) 53–76.

36. S. Ueno, Y. Kajitani and S. Gotoh, On the nonseparating independent set problem and feedback set problem for graphs with no vertex degree exceeding three, Discrete Mathematics 72 (1988) 355–360.

37. T. Wanatabe, S. Kajita and K. Onaga, Vertex covers and connected vertex covers in 3-connected graphs, Proc. IEEE International Symposium on Circuits and Systems 1991, 1017–1020.