Induced Disjoint Paths and Connected Subgraphs for H-Free Graphs

Barnaby Martin[2[0000−0002−4642−8614]], Daniël Paulusma[2[0000−0001−5945−9287]], Siani Smith[2✉[0000−0003−0797−0512]], and Erik Jan van Leeuwen[3[0000−0001−5240−7257]]

[1] Department of Computer Science, Durham University, Durham, UK, {barnaby.d.martin,daniel.paulusma,siani.smith}@durham.ac.uk

[2] Department of Information and Computing Sciences, Utrecht University, The Netherlands, e.j.vanleeuwen@uu.nl

Abstract. Paths $P^1, \ldots, P^k$ in a graph $G = (V, E)$ are mutually induced if any two distinct P^i and P^j have neither common vertices nor adjacent vertices. The Induced Disjoint Paths problem is to decide if a graph G with k pairs of specified vertices (s_i, t_i) contains k mutually induced paths P^i such that each P^i starts from s_i and ends at t_i. This is a classical graph problem that is NP-complete even for $k = 2$. We introduce a natural generalization, Induced Disjoint Connected Subgraphs: instead of connecting pairs of terminals, we must connect sets of terminals. We give almost-complete dichotomies of the computational complexity of both problems for H-free graphs, that is, graphs that do not contain some fixed graph H as an induced subgraph. Finally, we give a complete classification of the complexity of the second problem if the number k of terminal sets is fixed, that is, not part of the input.

Keywords: induced subgraphs · connectivity · H-free graph · complexity dichotomy

1 Introduction

The well-known Disjoint Paths problem is one of the problems in Karp's list of NP-complete problems. It is to decide if a graph has pairwise vertex-disjoint paths $P^1, \ldots, P^k$ where each P^i connects two pre-specified vertices s_i and t_i. Its generalization, Disjoint Connected Subgraphs, plays a crucial role in the graph minor theory of Robertson and Seymour. This problem asks for connected subgraphs $D^1, \ldots, D^k$, where each D^i connects a pre-specified set of vertices Z_i. In a recent paper [18] we classified, subject to a small number of open cases, the complexity of both these problems for *H-free* graphs, that is, for graphs that do not contain some fixed graph H as an *induced* subgraph.

Our Focus. We consider the *induced* variants of Disjoint Paths and Disjoint Connected Subgraphs. These problems behave differently. Namely, Disjoint Paths for fixed k, or more generally, Disjoint Connected Subgraphs, after fixing both k and $\ell = \max\{|Z_1|, \ldots, |Z_k|\}$, is polynomial-time solvable [30]. In contrast, Induced Disjoint Paths is NP-complete even when $k = 2$, as shown both by Bienstock [2] and Fellows [5]. Just as for the classical problems [18], we perform a systematic study and focus on H-free graphs. As it turns out, for the restriction to H-free graphs, the induced variants actually become computationally easier for an infinite family of graphs H. We first give some definitions.

Terminology. For a subset $S \subseteq V$ in a graph $G = (V, E)$, let $G[S]$ denote the *induced* subgraph of G by S, that is, $G[S]$ is the graph obtained from G after removing every vertex not in S. Let $G_1 + G_2$ be the disjoint union of two vertex-disjoint graphs G_1 and G_2. We say that paths $P^1, \ldots, P^k$, for some $k \geq 1$, are *mutually induced paths* of G if there exists a set $S \subseteq V$ such that $G[S] = P^1 + \ldots + P^k$; note that every P^i is an induced path and that there is no edge between two vertices from different paths P^i and P^j. A path P is an *s-t-path* (or t-s-path) if the end-vertices of P are s and t.

A *terminal pair* (s, t) is an unordered pair of two distinct vertices s and t in a graph G, which we call *terminals*. A set $T = \{(s_1, t_1), \ldots, (s_k, t_k)\}$ of terminal pairs of G is a *terminal pair collection* if the terminals pairs are pairwise disjoint, so, apart from $s_i \neq t_i$ for $i \in \{1, \ldots, k\}$, we also have $\{s_i, t_i\} \cap \{s_j, t_j\} = \emptyset$ for every $1 \leq i < j \leq k$. We now define the following decision problem:

Induced Disjoint Paths
Instance: a graph G and terminal pair collection $T = \{(s_1, t_1) \ldots, (s_k, t_k)\}$.
Question: does G have a set of mutually induced paths P^1,…,P^k such that P^i is an s_i-t_i path for $i \in \{1, \ldots, k\}$?

Note that as every path between two vertices s and t contains an induced path between s and t, the condition that every P^i must be induced is not strictly needed in the above problem definition. We say that the paths $P^1, \ldots, P^k$, if they exist, form a *solution* of Induced Disjoint Paths.

We now generalize the above notions from pairs and paths to sets and connected subgraphs. Subgraphs $D^1, \ldots, D^k$ of a graph $G = (V, E)$ are *mutually induced subgraphs* of G if there exists a set $S \subseteq V$ such that $G[S] = D^1 + \ldots + D^k$. A connected subgraph D of G is a *Z-subgraph* if $Z \subseteq V(D)$. A *terminal set* Z is an unordered set of distinct vertices, which we again call *terminals*. A set $\mathcal{Z} = \{Z_1, \ldots, Z_k\}$ is a *terminal set collection* if $Z_1, \ldots, Z_k$ are pairwise disjoint terminal sets. We now introduce the generalization:

Induced Disjoint Connected Subgraphs
Instance: a graph G and terminal set collection $\mathcal{Z} = \{Z_1, \ldots, Z_k\}$.
Question: does G have a set of mutually induced connected subgraphs $D^1, \ldots, D^k$ such that D^i is a Z_i-subgraph for $i \in \{1, \ldots, k\}$?

The subgraphs $D^1, \ldots, D^k$, if they exist, form a *solution*. We write Induced Disjoint Connected ℓ-Subgraphs if $\ell = \max\{|Z_1|, \ldots, |Z_k|\}$ is fixed. Note that Induced Disjoint Connected 2-Subgraphs is exactly Induced Disjoint Paths.

1.1 Known Results

Only results for Induced Disjoint Paths are known and these hold for a slightly more general problem definition (see Section 6). Namely, Induced Disjoint Paths is linear-time solvable for circular-arc graphs [10]; polynomial-time solvable for chordal graphs [1], AT-free graphs [11], graph classes of bounded mim-width [15]; and NP-complete for claw-free graphs [6], line graphs of triangle-free chordless graphs [29] and thus for (theta,wheel)-free graphs, and for planar graphs; the last result follows from a result of Lynch [23] (see [11]). Moreover, Induced Disjoint Paths is XP with parameter k for (theta,wheel)-free graphs [29] and even FPT with parameter k for claw-free graphs [9] and planar graphs [17]; the latter can be extended to graph classes of bounded genus [20].

1.2 Our Results

Let P_r be the path on r vertices. A *linear forest* is the disjoint union of one or more paths. We write $F \subseteq_i G$ if F is an induced subgraph of G and sG for the disjoint union of s copies of G. We can now present our first two results: the first one includes our dichotomy for Induced Disjoint Paths (take $\ell = 2$).

Theorem 1. *Let $\ell \geq 2$. For a graph H,* Induced Disjoint Connected ℓ-Subgraphs *on H-free graphs is polynomial-time solvable if $H \subseteq_i sP_3 + P_6$ for some $s \geq 0$;* NP*-complete if H is not a linear forest; and quasipolynomial-time solvable otherwise.*

Theorem 2. *For a graph H such that $H \neq sP_1 + P_6$ for some $s \geq 0$,* Induced Disjoint Connected Subgraphs *on H-free graphs is polynomial-time solvable for H-free graphs if $H \subseteq_i sP_1 + P_3 + P_4$ or $H \subseteq_i sP_1 + P_5$ for some $s \geq 0$, and it is* NP*-complete otherwise.*

Note the complexity jumps if we no longer fix ℓ. We will show that all open cases in Theorem 2 are equivalent to exactly **one** open case, namely $H = P_6$.

Comparison. The Disjoint Connected Subgraphs problem restricted to H-free graphs is polynomial-time solvable if $H \subseteq_i P_4$ and else it is NP-complete, even if the maximum size of the terminal sets is $\ell = 2$, except for the three unknown cases $H \in \{3P_1, 2P_1{+}P_2, P_1{+}P_3\}$ [18]. Perhaps somewhat surprisingly, Theorems 1 and 2 show the induced variant is computationally easier for an infinite number of linear forests H (if P $\neq$ NP).

Fixing *k*. If the number k of terminal sets is fixed, we write k-Induced Disjoint Connected Subgraphs and prove the following complete dichotomy.

Theorem 3. *Let $k \geq 2$. For a graph H, k-*Induced Disjoint Connected Subgraphs *on H-free graphs is polynomial-time solvable for H-free graphs if $H \subseteq_i sP_1 + 2P_4$ or $H \subseteq_i sP_1 + P_6$ for some $s \geq 0$, and it is* NP*-complete otherwise.*

Comparison. We note a complexity jump between Theorems 2 and 3 when $H = sP_1 + 2P_4$ for some $s \geq 0$.

Paper Outline. Section 2 contains terminology, known results and auxiliary results that we will use as lemmas. Hardness results for Theorem 1 transfer to Theorem 2, whereas the reverse holds for polynomial results. As such, we show all our polynomial-time algorithms in Section 3 and all our hardness reductions in Section 4. The cases $H = sP_3 + P_6$ in Theorem 1 and $H = sP_1 + P_5$ in Theorem 2 are proven by a reduction to Independent Set via so-called *blob graphs*, just as the quasipolynomial-time result if H is a linear forest. Hence, we also include the proof of the latter result in Section 3. In Section 5 we combine the results from the previous two sections to prove Theorems 1–3.

In our theorems we have infinite families of polynomial cases related to nearly H-free graphs. For a graph H, a graph G is *nearly H-free* if G is $(P_1 + H)$-free. It is easy to see (cf [3]) that Independent Set is polynomial-time solvable on nearly H-free graphs if it is so on H-free graphs. However, for many other graph problems, this might either not be true or less easy to prove (see, for example, [16]). In Section 3 we show that it holds for the relevant cases in Theorem 2, in particular for the case $H = P_6$ (see Lemma 7). The latter result yields no algorithm but shows that essentially $H = P_6$ is the only one open case left in Theorem 2.

In Section 6 we consider a number of directions for future work. In particular we consider the restriction k-Disjoint Connected ℓ-Subgraphs where *both* k and ℓ are fixed and discuss some open problems.

2 Preliminaries

Let $G = (V, E)$ be a graph. A subset $S \subseteq V$ is *connected* if $G[S]$ is connected. A subset $D \subseteq V(G)$ is *dominating* if every vertex of $V(G) \setminus D$ is adjacent to least one vertex of D; if $D = \{v\}$ then v is a *dominating* vertex. The *open* and *closed neighbourhood* of a vertex $u \in V$ are $N(u) = \{v \mid uv \in E\}$ and $N[u] = N(u) \cup \{u\}$. For a set $U \subseteq V$ we define $N(U) = \bigcup_{u \in U} N(u) \setminus U$ and $N[U] = N(U) \cup U$.

For a graph $G = (V, E)$ and a subset $S \subseteq U$, we write $G - S = G[V \setminus S]$. If $S = \{u\}$ for some $u \in V$, we write $G - u$ instead of $G - \{u\}$. A vertex u is a *cut-vertex* of a connected graph G if $G - u$ is disconnected.

The *contraction* of an edge $e = uv$ in a graph G replaces the vertices u and v by a new vertex w that is adjacent to every vertex previously adjacent to u or v; note that the resulting graph G/e is still *simple*, that is, G/e contains no multi-edges or self-loops. The following lemma is easy to see (see, for example, [19]).

Lemma 1. *For a linear forest H, let G be an H-free graph. Then G/e is H-free for every $e \in E(G)$.*

In a solution $(D^1, \ldots, D^k)$ for an instance $(G, \mathcal{Z})$ of Induced Disjoint Connected Subgraphs, if D^i is minimal and X_i is a minimum connected dominating set of D^i, then $X_i \cup Z_i = D^i$ or, equivalently, $D^i \setminus X_i \subseteq Z_i$. This will be relevant in our proofs, where we use the following result of Camby and Schaudt, in particular for the case $r = 6$ (alternatively, we could use the slightly weaker characterization of P_6-free graphs in [13] but the below characterization gives a faster algorithm).

Theorem 4 ([4]). *Let $r \geq 4$ and G be a connected P_r-free graph. Let X be any minimum connected dominating set of G. Then $G[X]$ is either P_{r-2}-free or isomorphic to P_{r-2}.*

Let $G = (V, E)$ be a graph. Two sets $X_1, X_2 \subseteq V$ are *adjacent* if $X_1 \cap X_2 \neq \emptyset$ or there exists an edge with one end-vertex in X_1 and the other in X_2. The *blob graph* G° of G has vertex set $\{X \subseteq V(G) \mid X \text{ is connected}\}$ and edge set $\{X_1X_2 \mid X_1 \text{ and } X_2 \text{ are adjacent}\}$. Note that blob graphs may have exponential size, but in our proofs we will only construct parts of blob graphs that have polynomial size. We need the following known lemma that generalizes a result of Gartland et al. [8] for paths.

Lemma 2 ([27]). *For every linear forest H, a graph G is H-free if and only if G° is H-free.*

The Independent Set problem is to decide if a graph G has an *independent set* (set of pairwise non-adjacent vertices) of size at least k for some given integer k. We need the following two known results for Independent Set. The first one is due to Grzesik, Klimosová, Pilipczuk and Pilipczuk [12]. The second one is due to Pilipczuk, Pilipczuk and Rzążewski [28], who improved the previous quasipolynomial-time algorithm for Independent Set on P_t-free graphs, due to Gartland and Lokshtanov [7] (whose algorithm runs in $n^{O(\log^3 n)}$ time).

Theorem 5 ([12]). *The* Independent Set *problem is polynomial-time solvable for P_6-free graphs.*

Theorem 6 ([7]). *For every $r \geq 1$, the* Independent Set *problem can be solved in $n^{O(\log^2 n)}$ time for P_r-free graphs.*

Two instances of a decision problem are *equivalent* if one is a yes-instance if and only if the other one is. We frequently use the following lemmas (proofs omitted).

Lemma 3. *From an instance $(G, \mathcal{Z})$ of* Induced Disjoint Connected Subgraphs *we can in linear time, either find a solution for $(G, \mathcal{Z})$ or obtain an equivalent instance $(G', \mathcal{Z}')$ with $|V(G')| \leq |V(G)|$, such that the following holds:*

1. *$|\mathcal{Z}'| \geq 2$;*
2. *every $Z'_i \in \mathcal{Z}'$ has size at least 2; and*
3. *the union of the sets in $\mathcal{Z}'$ is an independent set.*

Moreover, if G is H-free for some linear forest H, then G' is also H-free.

Lemma 4. *Let H be a linear forest. If $(G, \mathcal{Z})$ is a yes-instance of* Induced Disjoint Connected Subgraphs *and G is H-free, then $(G, \mathcal{Z})$ has a solution $(D^1, \ldots, D^k)$, where each D^i has size at most $(2|V(H)| - 1)|Z_i|$.*

3 Algorithms

In this section we show all the polynomial-time and quasipolynomial-time results needed to prove our main theorems. We start with the following result.

Lemma 5. *Let $\ell \geq 2$. For every $s \geq 0$,* Induced Disjoint Connected ℓ-Subgraphs *is polynomial-time solvable for $(sP_3 + P_6)$-free graphs.*

Proof. Let $(G, \mathcal{Z})$ be an instance of the Induced Disjoint Connected ℓ-Subgraphs problem, where G is $(sP_3 + P_6)$-free for some $s \geq 0$. By Lemma 3, we may assume the union of the sets in $\mathcal{Z} = \{Z_1, \ldots, Z_k\}$ is independent.

First suppose that $k \leq s$. By Lemma 4 we may assume that each D^i in a solution $(D^1, \ldots, D^k)$ has size at most $t = (6s+11)\ell$. So $|D^1|+\ldots+|D^k|$ has size at most $kt \leq st$. Hence, we can consider all $O(n^{st})$ options of choosing a solution. As s and t are constants, this takes polynomial time in total. Now suppose that $k \geq s+1$. We consider all $O(n^{(s-1)t})$ options of choosing the first s subgraphs D^i, discarding those with an edge between distinct D^i or between some D^i and Z_j for some $j \geq s+1$. For each remaining option, let $G' = G - N[V(D^1) \cup \cdots \cup V(D^s)]$ and $\mathcal{Z}' = \{Z_{s+1}, \ldots, Z_k\}$. Note that G' is P_6-free.

Let F be the subgraph of the blob graph G'° induced by all connected subsets X in G' that have size at most 11ℓ, such that X contains all vertices of one set from $\mathcal{Z}'$ and no vertices from any other set of $\mathcal{Z}'$. Then F has polynomial size, as it has $O(n^{11\ell})$ vertices, so we can construct F in polynomial time. By Lemma 2, F is P_6-free.

We claim that $(G', \mathcal{Z}')$ has a solution if and only if F has an independent set of size $k - s$. First suppose that $(G', \mathcal{Z}')$ has a solution. Then, by Lemma 4, it has a solution $(D^{s+1}, \ldots, D^k)$, where each D^i has size at most 11ℓ. Such a solution corresponds to an independent set of size $k - s$ in F. For the reverse implication, two vertices in F that each contain vertices of the same set Z_i are adjacent. Hence, an independent set of size $k - s$ in F is a solution for $(G', \mathcal{Z}')$.

Due to the above, it remains to apply Theorem 5 to find in polynomial time whether G'° has an independent set of size $k - s$. ◻

By replacing Theorem 5 by Theorem 6 in the above proof and repeating the arguments of the second part we obtain the following result.

Lemma 6. *Let $\ell \geq 2$. For every $r \geq 1$,* Induced Disjoint Connected ℓ-Subgraphs *is quasipolynomial-time solvable for P_r-free graphs.*

We no prove a crucial lemma on nearly H-free graphs.

Lemma 7. *For $k \geq 2$, $r \leq 6$ and $s \geq 1$, if* (k-)Induced Disjoint Connected Subgraphs *is polynomial-time solvable for P_r-free, graphs, then it is so for $(sP_1 + P_r)$-free graphs.*

Proof. First let $r = 6$ and k be part of the input. Let $(G, \mathcal{Z})$ be an instance of Induced Disjoint Connected Subgraphs, where G is an $(sP_1 + P_6)$-free graph for some integer $s \geq 1$ and $\mathcal{Z} = \{Z_1, \ldots, Z_k\}$. We may assume without loss of generality that $|Z_1| \geq |Z_2| \geq \cdots \geq |Z_k|$. By Lemma 3, we may assume that $k \geq 2$; every $Z_i \in \mathcal{Z}$ has size at least 2; and the union of the sets in $\mathcal{Z}$ is an independent set. We assume that Induced Disjoint Connected Subgraphs is polynomial-time solvable for P_6-free graphs.

Case 1. For every $i \geq 2$, $|Z_i| \leq s-1$.
Let $D^1, \ldots, D^k$ be a solution for $(G, \mathcal{Z})$ (assuming it exists). By Lemma 4, we may assume without loss of generality that for $i \geq 2$, the number of vertices of D^i is at most $(2s+11)|Z_i| \leq (2s+11)(s-1)$.

First assume $k \leq s$. Then $V(D^2) \cup \cdots \cup V(D^k)$ has size at most t, where $t = (s-1)(2s+11)(s-1)$ is a constant. Hence, we can do as follows. We consider all $O(n^t)$ options for choosing the subgraphs $D^2, \ldots, D^k$. For each choice we check in polynomial time if $D^2, \ldots, D^k$ are mutually induced and connected, and if each D^i contains Z^i. We then check in polynomial time if the graph $G - N[(V(D^2) \cup \cdots V(D^k)]$ has a connected component containing Z_1. As the number of choices is polynomial, the total running time is polynomial.

Now assume $k \geq s+1$. We consider all $O(n^{s(2s+11)(s-1)})$ options of choosing the s subgraphs $D^2, \ldots, D^{s+1}$. We discard an option if for some $i \in \{1, \ldots, s\}$, the graph D^i is disconnected. We also discard an option if there is an edge between two vertices from two different subgraphs D^h and D^i for some $2 \leq h < i \leq s+1$, or if there is an edge between a vertex from some subgraph D^h $(2 \leq h \leq s)$ and a vertex from some set Z_i ($i = 1$ or $i \geq s+2$). If we did not discard the option, then we solve Induced Disjoint Connected Subgraphs on instance $(G - \bigcup_{i=2}^{s+2} N[V(D^i)], \mathcal{Z} \setminus \{Z_2, \ldots, Z_{s+1}\})$. The latter takes polynomial time as $G - \bigcup_{i=2}^{s+1} N[D^i]$ is P_6-free. As the number of branches is polynomial as well, the total running time is polynomial.

Case 2. $|Z_2| \geq s$ (and thus also $|Z_1| \geq s$).
Let $D^1, \ldots, D^k$ be a solution for $(G, \mathcal{Z})$ (assuming it exists). As $|Z_1| \geq s$, we find that for every $i \geq 2$, D^i is P_6-free. As $|Z_2| \geq s$, we also find that D^1 is P_6-free. Then, by setting $r = 6$ in Theorem 4, every D^i ($i \in \{1, \ldots, k\}$) has a connected dominating set X_i such that $G[X_i]$ is either P_4-free or isomorphic to P_4. We may assume that every X_i is inclusion-wise minimal (as else we could just replace X_i by a smaller connected dominating set of D^i).

Case 2a. There exist some X_i with size at least $7s+2$.
As $s \geq 1$, we have that $G[X_i]$ is P_4-free. We now set $r = 4$ in Theorem 4 and find that $G[X_i]$ has a connected dominating set Y_i of size at most 2. Hence, $G[X_i]$ contains a set R of $7s$ vertices that are not cut-vertices of $G[X_i]$. As X_i is minimal, this means that in D^i, each $r \in R$ has at least one neighbour $z \in Z_i$ that is not adjacent to any vertex of $X_i \setminus \{r\}$. We say that z is a *private* neighbour of r. We now partition R into sets $R_1, \ldots, R_7$, each of exactly s vertices. For $h = 1, \ldots, 7$, let $R_h = \{r_h^1, \ldots, r_h^s\}$ and pick a private neighbour z_h^j of r_h^j. For $h = 1, \ldots, 7$, let $Q_h = \{z_h^1, \ldots, z_h^s\}$. Each Q_h is independent, as Z_i is independent and $Q_h \subseteq Z_i$.

We claim that there exists an index $h \in \{1, \ldots, 7\}$ such that $G - (N[Q_h] \setminus R_h)$ is P_6-free. For a contradiction, assume that for every $h \in \{1, \ldots, 7\}$, we have that $G - (N[Q_h] \setminus R_h)$ is not P_6-free. As G is $(sP_1 + P_6)$-free and every Q_h is an independent set of size s, we have that $G - N[Q_h]$ is P_6-free. We conclude that every induced P_6 of G contains a vertex of R_h for every $h \in \{1, \ldots, 7\}$. This is contradiction, as every induced P_6 only has six vertices. Hence, there exists an index $h \in \{1, \ldots, 7\}$ such that $G - (N[Q_h] \setminus R_h)$ is P_6-free.

We exploit the above structural claim algorithmically as follows. We consider all $k = O(n)$ options that one of the sets X_i has size at least $7s + 2$. For each choice of index i we do as follows. We consider all $O(n^{2s})$ options of choosing a set Q_h of s vertices from the independent set Z_i together with a set R_h of s vertices from $N(Q_h)$. We discard the option if a vertex of Q_h has more than one neighbour in R_h, or if $G' = G - (N[Q_h] \setminus R_h)$ is not P_6-free. Otherwise, we solve Induced Disjoint Connected Subgraphs on instance $(G', \mathcal{Z}')$, where $\mathcal{Z}' = (\mathcal{Z} \setminus \{Z_i\}) \cup \{(Z_i \setminus Q_h) \cup R_h\}$. As G' is P_6-free, the latter takes polynomial time by our initial assumption. Hence, as the total number of branches is $O(n^{2s+1})$ the total running time of this check takes polynomial time.

Case 2b. Every X_i has size at most $7s + 1$.
First assume $k \leq s$. We consider all $O(n^{s(7s+1)})$ options of choosing the sets $X_1, \ldots, X_k$. For each option we check if $(X_1 \cup Z_1, \ldots, X_k \cup Z_k)$ is a solution for $(G, \mathcal{Z})$. As the latter takes polynomial time and the total number of branches is polynomial, this takes polynomial time.

Now assume $k \geq s + 1$. We consider all $O(n^{s(7s+1)})$ options of choosing the first s sets $X_1, \ldots, X_s$. We discard an option if for some $i \in \{1, \ldots, s\}$, the set $X_i \cup Z_i$ is disconnected. We also discard an option if there is an edge between two vertices from two different sets $X_h \cup Z_h$ and $X_i \cup Z_i$ for some $1 \leq h < i \leq s$, or if there is an edge between a vertex from some set $X_h \cup Z_h$ $(h \leq s)$ and a vertex from some set Z_i $(i \geq s + 1)$. If we did not discard the option, then we solve Induced Disjoint Connected Subgraphs on instance $(G - \bigcup_{i=1}^{s} N[X_i \cup Z_i], \{Z_{s+1}, \ldots, Z_k\})$. The latter takes polynomial time as $G - \bigcup_{i=1}^{s} N[X_i \cup Z_i]$ is P_6-free. As the number of branches is polynomial as well, the total running time is polynomial.

From the above case analysis we conclude that the running time of our algorithm is polynomial. If $r \leq 5$ and/or k is fixed we use exactly the same arguments. $\square$

Remark 1. Due to Lemma 7, the missing cases $H = sP_1 + P_6$ in Theorem 2 are all equivalent to the case $H = P_6$.

We will use Lemma 7 for the case where $r = 5$. We also make use of the blob approach again.

Lemma 8. *For every $s \geq 0$,* Induced Disjoint Connected Subgraphs *is polynomial-time solvable for $(sP_1 + P_5)$-free graphs.*

Proof. Due to Lemma 7 it suffices to prove the statement for P_5-free graphs only. Let $(G, \mathcal{Z})$ be an instance of Induced Disjoint Connected Subgraphs, where G is a P_5-free graph and $\mathcal{Z} = \{Z_1, \ldots, Z_k\}$. By Lemma 3, we may assume that $k \geq 2$; every $Z_i \in \mathcal{Z}$ has size at least 2; and the union of the sets in $\mathcal{Z}$ is an independent set. We may also delete every vertex from G that is not in a terminal set from $\mathcal{Z}$ but that is adjacent to two terminals in different sets Z_h and Z_i (such a vertex cannot be used in any subgraph of a solution). We now make a structural observation that gives us a procedure for safely contracting edges; recall that edge contraction preserves P_5-freeness by Lemma 1.

Consider a solution $(D^1 \dots D^k)$ that is *maximal* in the sense that any vertex v outside $V(D^1) \cup \cdots \cup V(D^k)$ must have a neighbour in at least two distinct subgraphs D^i and D^j. As G is P_5-free, v must be adjacent to all vertices of at least one of D^i and D^j. As v has no neighbours in both $Z_i \subseteq V(D^i)$ and $Z_j \subseteq V(D^j)$, v must be adjacent to all vertices of exactly one of D^i and D^j.

The above gives rise to the following algorithm. Let v be a vertex that is adjacent to at least one vertex $z \in Z_i$ but not to all vertices of Z_i. As v is adjacent to z and z is in Z_i, it hold that v does not belong to any D^h with $h \neq i$ for every (not necessarily maximal) solution $(D^1, \dots, D^k)$. The observation from the previous paragraph tells us that if v is not in any D^h and $(D^1, \dots, D^k)$ is a maximal solution, then v must be adjacent to all vertices of some D^j. As v is adjacent to $z \in Z_i$, it holds by construction that v is not adjacent to any vertex of any $Z_h \subseteq V(D^h)$ with $h \neq i$. Hence, $i = j$ must hold. However, this is not possible, as we assumed that v is not adjacent to all vertices of $Z_i \subseteq V(D^i)$. Hence, we may assume without loss of generality that v belongs to D^i (should a solution exist). This means that we can safely contract the edge vz and put the resulting vertex in Z_i. Then we apply Lemma 3 again and also remove all common neighbours of vertices from Z_i and vertices from other sets Z_j. This takes polynomial time and the resulting graph has one vertex less. Hence, by applying this procedure exhaustively we have, in polynomial time, either solved the problem or obtained an equivalent but smaller instance.

Suppose the latter case holds. For simplicity we denote the obtained instance by $(G, \mathcal{Z})$ again, where G is a P_5-free graph and $\mathcal{Z} = \{Z_1, \dots, Z_k\}$ with $k \geq 2$. Due to our procedure, every $Z_i \in \mathcal{Z}$ has size at least 2; the union of the sets in $\mathcal{Z}$ is an independent set. Moreover, every non-terminal vertex is adjacent either to no terminal vertex or is adjacent to all terminals of exactly one terminal set. We let S be the set of vertices of the latter type. Observe that it follows from the preceding that only vertices of S need to be used for a solution.

We now construct the subgraph F of the blob graph G° that is induced by all connected subsets X of the form $X = Z_i \cup \{s\}$ for some $1 \leq i \leq k$ and $s \in S$. Note that F has $O(kn)$ vertices. Hence, constructing F takes polynomial time. Moreover, F is P_5-free due to Lemma 2. As in the proof of Lemma 5, we observe that $(G, \mathcal{Z})$ has a solution if and only if F has an independent set of size k. It now remains to apply (in polynomial time) Theorem 5. $\square$

We now show a stronger result when k is fixed (proof omitted).

Lemma 9. *For every $s \geq 0$, k-*Induced Disjoint Connected Subgraphs *is polynomial-time solvable for $(sP_1 + P_6)$-free graphs.*

We now present our final two polynomial-time algorithms (proofs omitted).

Lemma 10. *For every $k \geq 2$ and $s \geq 0$, k-*Induced Disjoint Connected Subgraphs *is polynomial-time solvable for $(sP_1 + 2P_4)$-free graphs.*

Lemma 11. *For every $s \geq 0$,* Induced Disjoint Connected Subgraphs *is polynomial-time solvable for $(sP_1 + P_3 + P_4)$-free graphs.*

4 NP-Completeness Results

In this section we present a number of NP-completeness results; we omitted all proofs except one. If $\ell = 2$, we write Induced Disjoint Paths instead of Induced Disjoint Connected ℓ-Subgraphs. The *girth* of a graph G that is not a forest is the length of a shortest cycle of G.

Lemma 12. *For every $g \geq 3$,* Induced Disjoint Paths *is* NP*-complete for graphs of girth at least g.*

Lemma 13. *For every $g \geq 3$,* 2-Induced Disjoint Connected Subgraphs *is* NP*-complete for graphs of girth at least g.*

The *line graph* $L(G)$ of a graph G has vertex set $\{v_e \mid e \in E(G)\}$ and an edge between v_e and v_f if and only if e and f are incident on the same vertex in G. The following two lemmas show NP-completeness for line graphs. Lemma 14 is due to Fiala et al. [6]. They consider a more general variant of Induced Disjoint Paths, but their reduction holds in our setting as well. Lemma 15 can be derived from the NP-completeness of 2-Disjoint Connected Subgraphs [14].

Lemma 14 ([6]). Induced Disjoint Paths *is* NP*-complete for line graphs.*

Lemma 15. 2-Induced Disjoint Connected Subgraphs *is* NP*-complete for line graphs.*

Finally, we show two lemmas for graphs without certain induced linear forests.

Lemma 16. 2-Induced Disjoint Connected Subgraphs *is* NP*-complete for $(3P_2, P_7)$-free graphs.*

Proof. We reduce from Not-All-Equal-3-Sat, known to be NP-complete [31]. Let $(\mathcal{X}, \mathcal{C})$ be an instance of Not-All-Equal-3-Sat containing n variables $x_1, \ldots, x_n$ and m clauses $C_1, \ldots, C_m$. We construct a graph G as follows. Let X be a clique of size n on vertices $v_1, \ldots, v_n$. Introduce a copy v'_i of each v_i in X. Call the new set X' and make it a clique. Add the edges $v_i v'_i$ for each v_i in X. Let C be an independent set of size m on vertices $c_1, \ldots, c_m$. Introduce a copy c'_j of each vertex c_j in C. Call the new set C' (and keep it an independent set). Now for all $1 \leq i \leq n$ and $1 \leq j \leq m$, add an edge $v_i c_j$ and an edge $v'_i c'_j$ if clause C_j contains variable x_i. Set $Z_1 = C$ and $Z_2 = C'$. Then, (G, Z_1, Z_2) is an instance of 2-Induced Disjoint Connected Subgraphs.

Observe that G is P_7-free. Indeed, let P be any longest induced path in G. Then P can contain at most two vertices from X and at most two vertices from

X'. If P contains at most one vertex from C and at most one vertex from C', then P has length at most $2+2+1+1=6$. On the other hand, if P contains two vertices from C or two vertices from C', then P has length at most 3.

We also observe that G is $3P_2$-free, as any P_2 must contain at least one vertex from X or from X', and X and X' are cliques. So we are done after proving the following claim: $(\mathcal{X}, \mathcal{C})$ is a yes-instance of Not-All-Equal-3-Sat if and only if (G, Z_1, Z_2) is a yes-instance of 2-Induced Disjoint Connected Subgraphs.

In the forward direction, let τ be a satisfying truth assignment. We put in A every vertex of X for which the corresponding variable is set to true. We put in A' every vertex of X' for which the corresponding variable is set to false. As each clause C_j contains at least one true variable, c_j is adjacent to a vertex in A. Similarly, each clause C_j contains at least one false variable, so each c'_j is adjacent to a vertex in A'. As X and X' are cliques, A and A' are cliques. Hence, $G[C \cup A]$ and $G[C' \cup A']$ are connected.

Now suppose there is an edge between a vertex of $C \cup A$ and a vertex of $C' \cup A'$. Then, by construction, this edge must be equal to some $v_i v'_i$, which means that v_i is in A and v'_i is in A', so x_i must be true and false at the same time, a contradiction. Hence, there exists no edge between a vertex from $C \cup A$ and a vertex from $C' \cup A'$. We conclude that $(C \cup A, C' \cup A')$ is a solution.

In the backwards direction, let $(C \cup A, C' \cup A')$ be a solution. Then, by definition, there is no edge between $C \cup A$ and $C' \cup A'$, which means that there is no edge between A and A'. Then $A \subseteq X$ and $A' \subseteq X'$, since X and X' are cliques and A (A') needs to contain at least one vertex of X (X'). Also, there is no variable x_i such that v_i is in A and v'_i is in A'. This means we can define a truth assignment τ by setting all variables corresponding to vertices in A to be true, all variables corresponding to vertices in A' to be false, and all remaining vertices in $\mathcal{X}$ to be true (or false, it does not matter).

As C is an independent set and $C \cup A$ is connected, each c_j has a neighbour in A. So each C_j contains a true literal. As C' is an independent set and $C' \cup A'$ is connected, each c'_j has a neighbour in A'. So each C_j contains a false literal. Hence, τ is a satisfying truth assignment. This completes the proof. □

Lemma 17. Induced Disjoint Connected Subgraphs *is* NP*-complete for* $2P_4$*-free graphs.*

5 The Proofs of Theorems 1–3

We are now ready to prove Theorems 1–3, which we restate below.

Proof of Theorem 1. We prove the theorem for $\ell = 2$; extending the proof to $\ell \geq 3$ is trivial. If H contains a cycle C_s, then we use Lemma 12 by setting the girth to $g = s+1$. Suppose that H contains no cycle, that is, H is a forest. If H contains a vertex of degree at least 3, then we use Lemma 14, as in that case the class of H-free graphs contains the class of $K_{1,3}$-free graphs, which in turn contains the class of line graphs. In the remaining cases, H is a linear forest. If $H \subseteq_i sP_3 + P_6$ for some $s \geq 0$ we use Lemma 5. Else we use Lemma 6. □

Proof of Theorem 2. If H is not a linear forest, we use Theorem 1. Suppose H is a linear forest. If $H \subseteq_i sP_1 + P_5$ for some $s \geq 0$ we use Lemma 8. If $H \subseteq_i sP_1 + P_3 + P_4$ for some $s \geq 0$ we use Lemma 11. If $3P_2 \subseteq_i H$ or $P_7 \subseteq_i H$ we use Lemma 16. Otherwise $2P_4 \subseteq_i H$ and we use Lemma 17. □

Proof of Theorem 3. If H contains a cycle C_s, then we use Lemma 13 by setting the girth to $g = s+1$. Suppose that H contains no cycle, that is, H is a forest. If H contains a vertex of degree at least 3, then we use Lemma 15, as in that case the class of H-free graphs contains the class of $K_{1,3}$-free graphs, which in turn contains the class of line graphs. In the remaining cases, H is a linear forest. If $H \subseteq_i sP_1+P_6$ for some $s \geq 0$ we use Lemma 9. If $H \subseteq_i sP_1+2P_4$ for some $s \geq 0$ we use Lemma 10. Otherwise $3P_2 \subseteq_i H$ or $P_7 \subseteq_i H$ and we use Lemma 16. □

6 Future Work

Our results naturally lead to some open problems. First of all, can we find polynomial-time algorithms for the quasipolynomial cases in Theorem 1? This is a challenging task that is also open for Independent Set; note that we reduce to the latter problem to solve the case where $H = sP_1 + P_6$ for some $s \geq 0$.

We also recall that the case $H = P_6$ is essentially the only remaining open case left in Theorem 2, which is for the setting where k and ℓ are both part of the input. As shown in Theorems 1 and 3, respectively, we have a positive answer for the settings where ℓ is fixed (and k is part of the input) and where k is fixed (and ℓ is part of the input), respectively. However, it seems challenging to combine the techniques when both k and ℓ are part of the input.

We did not yet discuss the k-Induced Disjoint Connected ℓ-Subgraphs problem, which is the variant where both k and ℓ are fixed; note that if $\ell = 2$, then we obtain the k-Induced Disjoint Paths problem. The latter problem restricted to $k = 2$ is closely related to the problem of deciding if a graph contains a cycle passing through two specified vertices and has been studied for hereditary graph classes as well; see [21]. Recently, we made some more progress. A *subdivided claw* is obtained from a claw after subdividing each edge zero or more times. In particular, the *chair* is the graph obtained from the claw by subdividing one of its edges exactly once. The set $\mathcal{S}$ consists of all graphs with the property that each of their connected components is a path or a subdivided claw. We proved in [24] that for every integer $k \geq 2$ and graph H, k-Induced Disjoint Paths is polynomial-time solvable if H is a subgraph of the disjoint union of a linear forest and a chair, and it is NP-complete if H is not in $\mathcal{S}$.

From the above it follows in particular that k-Induced Disjoint Paths is polynomial-time solvable for claw-free graphs (just like Independent Set [26, 32]) in contrast to the other three variants, which are NP-complete for claw-free graphs (see Theorems 1–3). We leave completing the classification of k-Induced Disjoint Paths as future work and refer to [24] for a more in-depth discussion.

Acknowledgments. We thank Paweł Rzążewski for the argument using blob graphs, which simplified two of our proofs and led to the case $H = P_6$ in Theorem 1.

References

1. Belmonte, R., Golovach, P.A., Heggernes, P., van't Hof, P., Kaminski, M., Paulusma, D.: Detecting fixed patterns in chordal graphs in polynomial time. Algorithmica **69**, 501–521 (2014)
2. Bienstock, D.: On the complexity of testing for odd holes and induced odd paths. Discrete Mathematics **90**, 85–92 (1991)
3. Brandstädt, A., Hoàng, C.T.: On clique separators, nearly chordal graphs, and the maximum weight stable set problem. Theoretical Computer Science **389**, 295–306 (2007)
4. Camby, E., Schaudt, O.: A new characterization of P_k-free graphs. Algorithmica **75**, 205–217 (2016)
5. Fellows, M.R.: The Robertson-Seymour theorems: A survey of applications. Proc. AMS-IMS-SIAM Joint Summer Research Conference, Contemporary Mathematics **89**, 1–18 (1989)
6. Fiala, J., Kamiński, M., Lidický, B., Paulusma, D.: The k-in-a-Path problem for claw-free graphs. Algorithmica **62**, 499–519 (2012)
7. Gartland, P., Lokshtanov, D.: Independent set on P_k-free graphs in quasi-polynomial time. Proc. FOCS 2020 pp. 613–624 (2020)
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 pp. 330–341 (2021)
9. Golovach, P.A., Paulusma, D., van Leeuwen, E.J.: Induced disjoint paths in claw-free graphs. SIAM Journal on Discrete Mathematics **29**, 348–375 (2015)
10. Golovach, P.A., Paulusma, D., van Leeuwen, E.J.: Induced disjoint paths in circular-arc graphs in linear time. Theoretical Computer Science **640**, 70–83 (2016)
11. Golovach, P.A., Paulusma, D., van Leeuwen, E.J.: Induced disjoint paths in AT-free graphs. Journal of Computer and System Sciences **124**, 170–191 (2022)
12. Grzesik, A., Klimosová, T., Pilipczuk, M., Pilipczuk, M.: Polynomial-time algorithm for maximum weight independent set on P_6-free graphs. Proc. SODA 2019 pp. 1257–1271 (2019)
13. van't Hof, P., Paulusma, D.: A new characterization of P_6-free graphs. Discrete Applied Mathematics **158**, 731–740 (2010)
14. van't Hof, P., Paulusma, D., Woeginger, G.J.: Partitioning graphs into connected parts. Theoretical Computer Science **410**, 4834–4843 (2009)
15. Jaffke, L., Kwon, O., Telle, J.A.: Mim-width I. induced path problems. Discrete Applied Mathematics **278**, 153–168 (2020)
16. Johnson, M., Paesani, G., Paulusma, D.: Connected Vertex Cover for $(sP_1 + P_5)$-free graphs. Algorithmica **82**, 20–40 (2020)
17. Kawarabayashi, K., Kobayashi, Y.: A linear time algorithm for the induced disjoint paths problem in planar graphs. Journal of Computer and System Sciences **78**, 670–680 (2012)
18. Kern, W., Martin, B., Paulusma, D., Smith, S., van Leeuwen, E.J.: Disjoint paths and connected subgraphs for H-free graphs. Theoretical Computer Science **898**, 59–68 (2022)
19. Kern, W., Paulusma, D.: Contracting to a longest path in H-free graphs. Proc. ISAAC 2020, LIPIcs **181**, 22:1–22:18 (2020)
20. Kobayashi, Y., Kawarabayashi, K.: Algorithms for finding an induced cycle in planar graphs and bounded genus graphs. Proc. SODA 2009 pp. 1146–1155 (2009)

21. Lévêque, B., Lin, D.Y., Maffray, F., Trotignon, N.: Detecting induced subgraphs. Discrete Applied Mathematics **157**, 3540–3551 (2009)
22. Li, W.N.: Two-segmented channel routing is strong NP-complete. Discrete Applied Mathematics **78**, 291–298 (1997)
23. Lynch, J.: The equivalence of theorem proving and the interconnection problem. SIGDA Newsletter **5**, 31–36 (1975)
24. Martin, B., Paulusma, D., Smith, S., van Leeuwen, E.J.: Few induced disjoint paths for H-free graphs. Proc. ISCO 2022, LNCS, to appear
25. Martin, B., Paulusma, D., Smith, S., van Leeuwen, E.J.: Induced disjoint paths and connected subgraphs for H-free graphs. Proc. WG 2022, LNCS, to appear
26. Minty, G.J.: On maximal independent sets of vertices in claw-free graphs. Journal of Combinatorial Theory, Series B **28**, 284–304 (1980)
27. 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
28. Pilipczuk, M., Pilipczuk, M., Rzążewski, P.: Quasi-polynomial-time algorithm for independent set in P_t-free graphs via shrinking the space of induced paths. Proc. SOSA 2021 pp. 204–209 (2021)
29. Radovanović, M., Trotignon, N., Vušković, K.: The (theta,wheel)-free graphs Part IV: Induced paths and cycles. Journal of Combinatorial Theory, Series B **146**, 495–531 (2021)
30. Robertson, N., Seymour, P.D.: Graph minors .XIII. The Disjoint Paths problem. Journal of Combinatorial Theory, Series B **63**, 65–110 (1995)
31. Schaefer, T.J.: The complexity of satisfiability problems. STOC pp. 216–226 (1978)
32. Shibi, N.: Algorithme de recherche d'un stable de cardinalité maximum dans un graphe sans étoile. Discrete Mathematics **29**, 53–76 (1980)