Maximum List r-Colorable Induced Subgraphs in kP3-Free Graphs

Citation for the original published paper (version of record):
Galby, E., Lima, P., Munaro, A. et al (2025). Maximum List r-Colorable Induced Subgraphs in kP3-Free Graphs. Leibniz International Proceedings in Informatics, LIPIcs, 351.
http://dx.doi.org/10.4230/LIPIcs.ESA.2025.40

Maximum List r-Colorable Induced Subgraphs in kP_3-Free Graphs

Esther Galby
Chalmers University of Technology, Gothenburg, Sweden
University of Gothenburg, Sweden

Paloma T. Lima
IT University of Copenhagen, Denmark

Andrea Munaro
Department of Mathematical, Physical and Computer Sciences, University of Parma, Italy

Amir Nikabadi
IT University of Copenhagen, Denmark

Abstract

We show that, for every fixed positive integers r and k, Max-Weight List r-Colorable Induced Subgraph admits a polynomial-time algorithm on kP_3-free graphs. This problem is a common generalization of Max-Weight Independent Set, Odd Cycle Transversal and List r-Coloring, among others. Our result has several consequences.

First, it implies that, for every fixed $r \geq 5$, assuming $\mathsf{P} \neq \mathsf{NP}$, Max-Weight List r-Colorable Induced Subgraph is polynomial-time solvable on H-free graphs if and only if H is an induced subgraph of either kP_3 or $P_5 + kP_1$, for some $k \geq 1$. Second, it makes considerable progress toward a complexity dichotomy for Odd Cycle Transversal on H-free graphs, allowing to answer a question of Agrawal, Lima, Lokshtanov, Rzążewski, Saurabh, and Sharma [ACM Trans. Algorithms 2025]. Third, it gives a short and self-contained proof of the known result of Chudnovsky, Hajebi, and Spirkl [Combinatorica 2024] that List r-Coloring on kP_3-free graphs is polynomial-time solvable for every fixed r and k.

We also consider two natural distance-d generalizations of Max-Weight Independent Set and List r-Coloring and provide polynomial-time algorithms on kP_3-free graphs for every fixed integers r, k, and $d \geq 6$.

2012 ACM Subject Classification Mathematics of computing → Graph coloring; Theory of computation → Graph algorithms analysis

Keywords and phrases Hereditary classes, list coloring, odd cycle transversal, independent set

Digital Object Identifier 10.4230/LIPIcs.ESA.2025.40

Related Version *Full Version*: `https://arxiv.org/abs/2505.00412`

Funding *Paloma T. Lima*: Supported by Independent Research Fund Denmark grant agreement number 2098-00012B.
Amir Nikabadi: Supported by Independent Research Fund Denmark grant agreement number 2098-00012B.

1 Introduction

A fundamental class of graph optimization problems consists in finding a maximum-weight induced subgraph satisfying a certain fixed property Π. Lewis and Yannakakis [12] showed that, whenever this fixed property Π is nontrivial and hereditary, the corresponding problem is NP-hard. In this paper, we investigate the case where Π is the property of being list r-colorable, leading to a meta-problem called Max-Weight List r-Colorable Induced Subgraph. In order to properly define this problem, we first require some definitions.

33rd Annual European Symposium on Algorithms (ESA 2025).
Editors: Anne Benoit, Haim Kaplan, Sebastian Wild, and Grzegorz Herman; Article No. 40; pp. 40:1–40:13
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Let $G = (V, E)$ be a finite simple graph. A *coloring* of G is a mapping $\phi\colon V \to \{1, 2, \dots\}$ that gives each vertex $u \in V$ a *color* $\phi(u)$ in such a way that, for every two adjacent vertices u and v in G, we have that $\phi(u) \neq \phi(v)$. For $r \geq 1$, a coloring ϕ of G is an *r-coloring* if $\phi(u) \in \{1, \dots, r\}$ for every $u \in V$, and a graph is *r-colorable* if it admits an r-coloring. For $r \geq 1$, an *r-list assignment* of G is a function $L\colon V \to 2^{\{1,\dots,r\}}$ that assigns each vertex $u \in V$ a *list* $L(u) \subseteq \{1, \dots, r\}$ of admissible colors for u. A coloring ϕ of G *respects* L if $\phi(u) \in L(u)$ for every $u \in V$. We are finally ready to define Max-Weight List r-Colorable Induced Subgraph, where r is a fixed positive integer.

Max-Weight List r-Colorable Induced Subgraph

Input: A graph G equipped with a weight function $w\colon V(G) \to \mathbb{Q}_+$, and an r-list assignment L of G.
Task: Find a subset $F \subseteq V(G)$ such that:
1. The induced subgraph $G[F]$ admits a coloring that respects L, and
2. The weight $w(F) = \sum_{v \in F} w(v)$ is maximum subject to the condition above.

Max-Weight List r-Colorable Induced Subgraph is a common generalization of several well-known and deeply investigated NP-hard problems, as we explain next. Max-Weight List r-Colorable Induced Subgraph generalizes List r-Coloring and hence r-Coloring as well, which are known to be NP-hard for all $r > 2$ [11]. Recall that, for a fixed $r \geq 1$, List r-Coloring is the problem to decide whether a given graph G with an r-list assignment L admits a coloring that respects L. By setting $L(u) = \{1, \dots, r\}$ for every $u \in V(G)$, we obtain r-Coloring. Note also that, for $r_1 \leq r_2$, List r_1-Coloring is a special case of List r_2-Coloring.

Several other NP-hard problems are special cases of Max-Weight List r-Colorable Induced Subgraph for specific values of r. For example, for $r = 1$, Max-Weight List r-Colorable Induced Subgraph is equivalent to Max-Weight Independent Set, which is the problem of finding a maximum-weight subset of pairwise non-adjacent vertices of an input graph G. For $r = 2$, Max-Weight List r-Colorable Induced Subgraph generalizes the problem of finding a maximum-weight induced bipartite subgraph of an input graph G which, by complementation, is equivalent to finding a minimum-weight subset of vertices intersecting all odd cycles in G. The latter is the well-known Odd Cycle Transversal.

Given the hardness of Max-Weight List r-Colorable Induced Subgraph, it is natural to investigate whether the problem becomes tractable for restricted classes of inputs. The framework of hereditary graph classes (i.e., graph classes closed under vertex deletion) is particularly well suited for this type of research, where the ultimate goal is to obtain complexity dichotomies telling us for which hereditary graph classes the problem at hand can or cannot be solved efficiently (under the standard complexity assumption that $\mathsf{P} \neq \mathsf{NP}$).

We recall some relevant definitions. A graph G is *H-free*, for some graph H, if it contains no induced subgraph isomorphic to H. For a set of graphs $\{H_1, \dots, H_p\}$, a graph is *$(H_1, \dots, H_p)$-free* if it is H_i-free for every $i \in \{1, \dots, p\}$. The *disjoint union* $G + H$ of graphs G and H is the graph with vertex set $V(G) \cup V(H)$ and edge set $E(G) \cup E(H)$. We denote the disjoint union of k copies of G by kG and let P_s denote the chordless path on s vertices.

It is known that, for $r = 2$, Max-Weight List r-Colorable Induced Subgraph admits no polynomial-time algorithm on H-free graphs unless H is a *linear forest* (i.e., a disjoint union of paths). Indeed, Chiarelli et al. [3] showed that its special case Odd Cycle Transversal is NP-hard on H-free graphs if H contains a cycle or a claw (the claw is the 4-vertex star). In recent years, considerable work has been done toward classifying the complexity of Odd Cycle Transversal (and its generalizations) on graphs forbidding an induced linear forest. In Theorem 1, we collect known results for Odd Cycle Transversal and its two generalizations Max-Weight r-Colorable Induced Subgraph and Max-Weight List r-Colorable Induced Subgraph on H-free graphs, where H is a linear forest. The problems are listed in increasing order of generality. In particular, an NP-hardness result for a certain problem implies NP-hardness for a more general problem. Note also that the hardness results hold even in the unweighted setting.

▶ **Theorem 1.** *The following hold:*

(i) *Odd Cycle Transversal on H-free graphs can be solved in polynomial time if*
- $H = P_5$ (Agrawal et al. [1]), *or*
- $H = kP_2$ *for all* $k \in \mathbb{N}$ (Chiarelli et al. [3]), *or*
- $H = P_3 + kP_1$ *for all* $k \in \mathbb{N}$ (Dabrowski et al. [7]),

and remains NP-hard if
- $H = (P_6, P_5 + P_2)$ (Dabrowski et al. [7]).

(ii) *Max-Weight r-Colorable Induced Subgraph on H-free graphs can be solved in polynomial time if*
- $H = P_5 + kP_1$ *for all* $k \in \mathbb{N}$ (Henderson et al. [10]),

and remains NP-hard if
- $H = (P_6, P_5 + P_2)$ *for* $r = 2$, *or*
- $H = 2P_4$ *for all* $r \geq 5$ (Hajebi et al. [9]).

(iii) *Max-Weight List r-Colorable Induced Subgraph on H-free graphs can be solved in polynomial time if*
- $H = P_5$ (Lokshtanov et al. [13]),
- $H = P_5 + kP_1$ *for all* $k \in \mathbb{N}$ (Henderson et al. [10]),

and remains NP-hard if
- $H = (P_6, P_5 + P_2)$ *for* $r \geq 2$, *or*
- $H = P_4 + P_2$ *for all* $r \geq 5$ (Couturier et al. [6]).

Recently, Chudnovsky et al. [4] obtained the following complete complexity dichotomy for List r-Coloring when $r \geq 5$. Assuming P $\neq$ NP, List r-Coloring ($r \geq 5$) can be solved in polynomial time on H-free graphs if and only if H is an induced subgraph of either kP_3 or $P_5 + kP_1$, for some $k \in \mathbb{N}$. Their main result toward this was showing that List r-Coloring ($r \geq 1$) can be solved in polynomial time on kP_3-free graphs, for any $k \in \mathbb{N}$, and this was obtained building on a very technical result of Hajebi et al. [9, Theorem 5.1].

Motivated by the quest for a complexity dichotomy, Agrawal et al. [1] posed very recently as an open problem to classify the computational complexity of Odd Cycle Transversal on $(P_3 + P_2)$-free graphs, the unique minimal open case stemming from Theorem 1.

It should also be mentioned that classifying the complexity of Max-Weight Independent Set on H-free graphs when H is a linear forest is a notorious open problem in algorithmic graph theory (see [5] for the state of the art). Note however that Max-Weight Independent Set substantially differs from Odd Cycle Transversal on H-free graphs, in the sense that it is polynomial-time solvable on claw-free graphs.

In this paper, we also consider the distance-d generalizations of Max-Weight Independent Set and List r-Coloring, defined as follows. For $d \geq 2$, a *distance-d independent set* of a graph G is a set of vertices of G pairwise at distance at least d in G. For fixed $d \geq 2$, Max-Weight Distance-d Independent Set (also known as d-Scattered Set) is the problem to find a maximum-weight distance-d independent set of an input graph G.

Trivially, for every fixed $d \geq 2$, Max-Weight Distance-d Independent Set is easy on P_{d+1}-free graphs, and the following hardness results are known.

▶ **Theorem 2.** *The following hold for Distance-d Independent Set on H-free graphs:*

(i) *It is* NP*-hard if H contains $2P_{(d+1)/2}$, for every fixed odd $d \geq 3$* (Eto et al. [8])*;*

(ii) *Assuming ETH, it admits no $2^{o(|V(H)|+|E(H)|)}$-time algorithm if H contains a cycle or a claw, for every fixed $d \geq 3$* (Bacsó et al. [2])*.*

For $d \geq 2$, a (d, r)*-coloring* of a graph G is an assignment of colors to the vertices of G using at most r colors such that no two distinct vertices at distance less than d receive the same color. Thus, a $(2, r)$-coloring is nothing but an r-coloring. Similarly as above, for fixed $d \geq 2$ and $r \geq 1$, we define (d, r)-Coloring as the problem of determining whether a given graph G has a (d, r)-coloring. List (d, r)-Coloring is defined similarly but we require in addition that every vertex u must receive a color from some given list $L(u) \subseteq \{1, \ldots, r\}$.

Sharp [18] provided the following complexity dichotomy: For fixed $d \geq 3$, (d, r)-Coloring is polynomial-time solvable for $r \leq \lfloor 3d/2 \rfloor$ and NP-hard for $r > \lfloor 3d/2 \rfloor$.

Our results. We prove three main algorithmic results for kP_3-free graphs. The first result, proven in Section 3, concerns Max-Weight List r-Colorable Induced Subgraph.

▶ **Theorem 3.** *Let $r \geq 1$ be a fixed integer. For every $k \in \mathbb{N}$, Max-Weight List r-Colorable Induced Subgraph can be solved in polynomial time on kP_3-free graphs.*

Theorem 3 has several interesting consequences. First, it immediately implies that Odd Cycle Transversal can be solved in polynomial time on kP_3-free graphs, for every $k \in \mathbb{N}$, thus solving a generalized version of the aforementioned open problem of Agrawal et al. [1]. Our result for Odd Cycle Transversal also complements the polynomial-time algorithms for Feedback Vertex Set and Even Cycle Transversal on kP_3-free graphs of Paesani et al. [16]. Second, Theorem 3 generalizes the recent result of Chudnovsky et al. [4] that List r-Coloring can be solved in polynomial time on kP_3-free graphs for every $r, k \in \mathbb{N}$. Although partially inspired by their approach, as we explain below, our proof of the more general Theorem 3 has the advantage of being considerably shorter and self-contained.

Theorem 3 also makes considerable progress toward a complete complexity dichotomy for Max-Weight List r-Colorable Induced Subgraph and Odd Cycle Transversal on H-free graphs. Indeed, paired with the recent result of [10], it *completely* settles the complexity of Max-Weight List r-Colorable Induced Subgraph on H-free graphs for $r \geq 5$ (see Theorem 1 and the discussion preceding it):

▶ **Theorem 4.** *Let $r \geq 5$ be a fixed integer. Assuming* $\mathsf{P} \neq \mathsf{NP}$*, Max-Weight List r-Colorable Induced Subgraph on H-free graphs is polynomial-time solvable if and only if H is an induced subgraph of either kP_3 or $P_5 + kP_1$, for some $k \geq 1$.*

Moreover, paired with the results of [3, 7, 10] mentioned above, Theorem 3 leaves the case $H = k_4P_4 + k_3P_3 + k_2P_2 + k_1P_1$, with $k_4 \geq 1$ and $k_4 + k_3 \geq 2$, as the only remaining open case toward a complete complexity dichotomy for Odd Cycle Transversal on H-free graphs.

We then consider the distance-d generalizations of Max-Weight Independent Set and List r-Coloring, for $d \geq 6$, and prove the following two results.

▶ **Theorem 5.** *Let $d \geq 6$ be a fixed integer. For every $k \in \mathbb{N}$, Max-Weight Distance-d Independent Set can be solved in polynomial time on kP_3-free graphs.*

▶ **Theorem 6.** *Let $d \geq 6$ and $r \geq 1$ be fixed integers. For every $k \in \mathbb{N}$, List (d, r)-Coloring can be solved in polynomial time on kP_3-free graphs.*

Paired with the aforementioned result of Eto et al. [8], Theorem 5 completely settles the computational complexity of Max-Weight Distance-d Independent Set on kP_3-free graphs, except for the *only* remaining open case $d = 4$.

Technical overview. We now explain our approach toward Theorem 3, which combines ideas from [14, 15] and [4]. It is instructive to first consider the special case of Odd Cycle Transversal (by complementation, Max-Weight 2-Colorable Induced Subgraph) on kP_2-free graphs, where a very simple algorithm can be obtained. Indeed, kP_2-free graphs have polynomially many (inclusion-wise) maximal independent sets, and these can be enumerated in polynomial time. Consequently, a maximum-weight induced bipartite subgraph can be found by exhaustively enumerating all pairs of maximal independent sets [3]. However, it is easily seen that even P_3-free graphs (i.e., graphs whose connected components are cliques) do not have polynomially many maximal independent sets. But it turns out that a weaker property is enough for our purposes: Admitting a polynomial family of "well-behaved" vertex sets such that every maximal independent set is contained in one of these sets. In our case, "well-behaved" means inducing a P_3-free subgraph. The intuition is that, given such a family, we can efficiently guess the color classes, each of which will be a disjoint union of cliques, and then match vertices to the possible color classes.

The following key notion, which we dub amiable family, was first introduced by Lozin and Mosca [15]. For a graph G, a family $\mathcal{S} \subseteq 2^{V(G)}$ of subsets of $V(G)$ is an *amiable family* if it satisfies the following two properties:

- Each member of $\mathcal{S}$ induces a P_3-free subgraph in G;
- Each (inclusion-wise) maximal independent set of G is contained in some member of $\mathcal{S}$.

Lozin and Mosca [15] showed that, when $k = 2$, every kP_3-free graph G admits an amiable family of size polynomial in $|V(G)|$ and which can be computed in polynomial time. Later, Lozin [14] observed how such property in fact holds for every $k \geq 2$ (see Lemma 8 for a formal statement). Given an amiable family $\mathcal{S}$ of polynomial size of a kP_3-free graph G, we would like to exhaustively solve Max-Weight List r-Colorable Induced Subgraph on every possible r-tuple consisting of members of $\mathcal{S}$. More precisely, let $(S_1, \ldots, S_r) \in \mathcal{S}^r$ be an r-tuple of members of $\mathcal{S}$. We would like to find a maximum-weight induced subgraph of $G[\bigcup_{i \in [r]} S_i]$ which admits an r-coloring respecting the given r-list assignment and such that, for $i = 1, \ldots, r$, all vertices colored i are contained in S_i. To do this, we then extend an idea of Chudnovsky et al. [4] as follows. We reduce our problem to that of finding a maximum-weight matching in an auxiliary bipartite graph where one partition class Y consists of $\bigcup_{i \in [r]} S_i$, the other class X consists of the connected components of the subgraphs $G[S_i]$, for $i = 1, \ldots, r$, and there is an edge between $y \in Y$ and $x \in X$ if and only if y belongs to the connected component x. Since each weighted matching problem can be solved in polynomial time using the Hungarian method (see, e.g., [17, Theorem 17.3]) and we build $|\mathcal{S}|^r$ auxiliary problems, which is a polynomial in $|V(G)|$, a solution to Max-Weight List r-Colorable Induced Subgraph can be found in polynomial time.

In order to prove Theorems 5 and 6, we consider the following distance-d generalization of the notion of amiable family. For a graph G, a family $\mathcal{S} \subseteq 2^{V(G)}$ of subsets of $V(G)$ is a *distance-d amiable family* if it satisfies the following properties:

- Each member of $\mathcal{S}$ induces a P_3-free subgraph in G;
- For each $S \in \mathcal{S}$, the connected components of $G[S]$ are pairwise at distance at least d in G;
- Each (inclusion-wise) maximal distance-d independent set of G is contained in some member of $\mathcal{S}$.

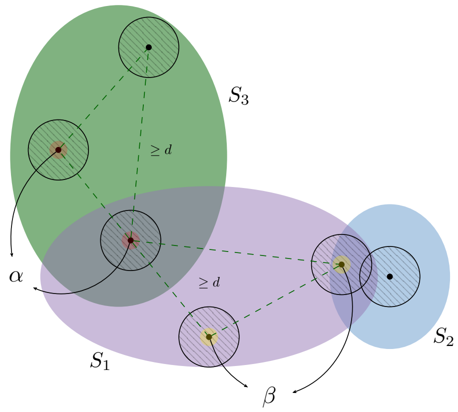

Figure 1 Visualization for distance-d amiable family $\mathcal{S} = \{S_1, S_2, S_3\}$. Circles represent cliques and α, β are maximal distance-d independent sets. Dashed lines depict paths of lengths at least d.

Clearly, a distance-2 amiable family is nothing but an amiable family. Our main technical contribution is the following Lemma 7, proven in Section 4. Although the algorithm for Lemma 7 is inspired by the case $d = 2$ and hence by the work of Lozin and Mosca [15], its proof of correctness is much more involved and requires genuinely new ideas.

▶ **Lemma 7.** *Let $d \geq 6$ be a fixed integer. For every $k \in \mathbb{N}$, every kP_3-free graph admits a distance-d amiable family of size $|V(G)|^{O(k)}$, which can be computed in time $|V(G)|^{O(k)}$.*

Equipped with Lemma 7, we immediately obtain Theorem 5. Indeed, in order to solve MAX-WEIGHT DISTANCE-d INDEPENDENT SET on a kP_3-free graph G, we simply find in polynomial time a distance-d amiable family $\mathcal{S}$ of G as above and, for each member $S \in \mathcal{S}$, find a max-weight independent set in the P_3-free graph $G[S]$. The latter can be clearly done in polynomial time, thus proving Theorem 5. The proof of Theorem 6 is similar to that of Theorem 3, the only difference being the use of a distance-d amiable family for $d \geq 6$, and hence omitted.

A remark about our approach, which combines ideas of Lozin and Mosca [14, 15] and Chudnovsky et al. [4], is in place. The proof of Chudnovsky et al. [4] that LIST r-COLORING ($r \geq 1$) is polynomial-time solvable on kP_3-free graphs generalizes the earlier result of Hajebi et al. [9] that LIST 5-COLORING is polynomial-time solvable on kP_3-free graphs, by replacing the second step in the proof of Hajebi et al. (see [9, Theorem 4.3]) with a significantly simpler argument (the aforementioned reduction to a bipartite matching problem) that, in addition, works for all $r \in \mathbb{N}$. However, their result still relies on the very technical first step of Hajebi et al. (see [9, Theorem 5.1]). Our approach can be viewed as a step further in the direction of simplifying and generalizing, as exemplified by our Theorems 3 and 6, which extend in different ways the main result of Chudnovsky et al. [4] by means of an arguably elegant and self-contained proof.

Proofs of statements marked with "$\star$" are omitted due to space constraints.

2 Preliminaries

We denote the set of positive integers by $\mathbb{N}$. For every $n \in \mathbb{N}$, we let $[n] := \{1, \ldots, n\}$. Given a set A, we denote by A^r the set of all ordered r-tuples of elements of A, i.e., $A^r = \{(a_1, \ldots, a_r) : a_i \in A \text{ for } i = 1, \ldots, r\}$.

All graphs in our paper are finite and simple. The *empty graph* is the graph with no vertices. Let now G be a graph. For $X \subseteq V(G)$, we denote the subgraph of G *induced* by X as $G[X]$, that is $G[X] = (X, \{uv : u, v \in X \text{ and } uv \in E(G)\})$. We use $N_G(X)$ to denote the neighbors in $V \setminus X$ of vertices in X. For disjoint sets $X, Y \subseteq V(G)$, we say that X is *complete* to Y if every vertex in X is adjacent to every vertex in Y, and X is *anticomplete* to Y if there are no edges between X and Y. For a subset $X \subseteq V(G)$, the *anti-neighborhood* of X, denoted by $\mathcal{A}(X)$, is the subset of vertices in $V(G) \setminus X$ which are anticomplete to X. With a slight abuse of notation, if $X = \{v_1, \ldots, v_i\}$, we denote $\mathcal{A}(X) = \mathcal{A}(\{v_1, \ldots, v_i\})$ by $\mathcal{A}(v_1, \ldots, v_i)$. Given two subsets $X, Y \subseteq V(G)$, an X,Y*-path* in G is a path in G which has one end in X, the other end in Y, and whose inner vertices belong to neither X nor Y. For vertices $u, v \in V(G)$, we denote by $d_G(u, v)$ the distance between u and v in G, i.e., the length of a shortest u,v-path in G. If no such path exists, we let $d_G(u, v) = \infty$. Moreover, for $d \in \mathbb{N}$, we let $N_G^{\geq d}(v) = \{u \in V(G) : d_G(v, u) \geq d\}$ and $N_G^{\leq d}(v) = \{u \in V(G) \setminus \{v\} : d_G(v, u) \leq d\}$. Given two subsets $X, Y \subseteq V(G)$, the *distance* between X and Y in G is the quantity $d_G(X, Y) = \min_{x \in X, y \in Y} d_G(x, y)$, i.e., the length of a shortest path in G between a vertex in X and a vertex in Y. Given $u \in V(G)$ and $Q \subseteq V(G)$, we say that u is connected to Q in G if there exists a u,Q-path in G (possibly of length 0).

A *clique* of a graph is a set of pairwise adjacent vertices and an *independent set* is a set of pairwise non-adjacent vertices. A *matching* of a graph is a set of pairwise non-adjacent edges. A *connected component* of G is a maximal connected subgraph of G. For convenience, we will often view a connected component as its vertex set rather than the subgraph itself. For this reason, we will say for example that "a connected component is a clique" rather than "a connected component is a complete subgraph". Throughout the paper, we will repeatedly make use of the fact that every connected component of a P_3-free graph is a clique.

3 The proof of Theorem 3

In this section we prove Theorem 3. As mentioned in the introduction, the first step is to obtain a polynomial-time algorithm for finding an amiable family (necessarily of polynomial size) of a kP_3-free graph. The second step consists then in reducing Max-Weight List r-Colorable Induced Subgraph to polynomially many auxiliary weighted matching problems. We do this in Lemma 9. We finally combine these two steps and prove Theorem 3.

▶ **Lemma 8** ($\star$)**.** *For every $k \in \mathbb{N}$, every kP_3-free graph G admits an amiable family of size $|V(G)|^{O(k)}$, which can be computed in time $|V(G)|^{O(k)}$.*

▶ **Lemma 9.** *Let $r \geq 1$ and $d \geq 2$ be fixed integers. Let G be a graph with weight function $w\colon V(G) \to \mathbb{Q}_+$, L an r-list assignment of G, and $\mathcal{S}$ a distance-d amiable family of G. Given an r-tuple $(S_1, \ldots, S_r) \in \mathcal{S}^r$, there exists an $O(((r+1)|V(G)|)^3)$-time algorithm that finds a maximum-weight induced subgraph H of $G[\bigcup_{i \in [r]} S_i]$ which admits a (d, r)-coloring $\phi\colon V(H) \to [r]$ satisfying the following:*
1. *For every $v \in V(H)$, $\phi(v) \in L(v)$;*
2. *For every $i \in [r]$, $\{v \in V(H)\colon \phi(v) = i\} \subseteq S_i$.*

Proof. Consider an r-tuple $(S_1, \ldots, S_r) \in \mathcal{S}^r$. For every $i \in [r]$, let c_i be the number of connected components of $G[S_i]$ and let $S_i^1, \ldots, S_i^{c_i}$ be an arbitrary ordering of the connected components of $G[S_i]$. By definition of distance-d amiable family, each such connected component of $G[S_i]$ is a clique and any two of them are pairwise at distance at least d in G.

The first step of our algorithm consists in preprocessing the graph $G[\bigcup_{i\in[r]} S_i]$ as follows. For every $i \in [r]$, if there exists a vertex $v \in S_i$ such that $i \notin L(v)$, then we remove v from S_i, that is, we set $S_i = S_i \setminus \{v\}$. Observe that this preprocessing is safe. Indeed, if H is an induced subgraph of $G[\bigcup_{i\in[r]} S_i]$ for which there exists a (d,r)-coloring ϕ satisfying both 1 and 2, then surely $\phi(v) \neq i$ if $v \in V(H)$.

We next show that finding such an induced subgraph of $G[\bigcup_{i\in[r]} S_i]$ of maximum weight boils down to finding a maximum-weight matching in an auxiliary weighted bipartite graph B constructed as follows. The graph B has bipartition $X \cup Y$ and edge set $E(B)$, where

$$X = \left\{x_{S_i^j} : i \in [r], j \in [c_i]\right\},\ Y = \left\{y_v : v \in \bigcup_{i\in[r]} S_i\right\} \text{ and } E(B) = \bigcup_{i=1}^{r}\bigcup_{j=1}^{c_i}\left\{y_v x_{S_i^j} : v \in S_i^j\right\}.$$

Moreover, for each $v \in \bigcup_{i\in[r]} S_i$, every edge of B incident to y_v is assigned the weight $w(v)$. Note that $|V(B)| = |\bigcup_{i\in[r]} S_i| + \sum_{i\in[r]} c_i \leq |V(G)| + r|V(G)| = (r+1)|V(G)|$.

▷ Claim 10 (⋆). Let $m \in \mathbb{Q}_+$. The graph $G[\bigcup_{i\in[r]} S_i]$ has an induced subgraph H with $w(V(H)) = m$ and which admits a (d,r)-coloring satisfying both 1 and 2 if and only if B has a matching of weight m.

Now, by Claim 10, our problem reduces to finding a maximum-weight matching in the bipartite graph B, which can be done in $O(|V(B)|^3)$-time using the Hungarian method (see [17, Theorem 17.3]). Since $|V(B)| \leq (r+1)|V(G)|$, this completes the proof of Lemma 9. ◀

We can finally sketch the proof of Theorem 3.

Proof of Theorem 3. Given a kP_3-free graph G and an r-list assignment L of G, the following algorithm outputs, in polynomial time, an induced subgraph H of G admitting a coloring that respects L and such that H is of maximum weight among all such subgraphs of G.

1. Compute an amiable family $\mathcal{S}$ of G of size $|V(G)|^{O(k)}$.
2. For each r-tuple $(S_1, \dots, S_r) \in \mathcal{S}^r$, find a maximum-weight induced subgraph H of $G[\bigcup_{i\in[r]} S_i]$ which admits a coloring $\phi\colon V(H) \to [r]$ satisfying 1 and 2 of Lemma 9.
3. Among all induced subgraphs computed in Step 2, output one of maximum weight. ◀

4 Distance-d amiable families: The proof of Lemma 7

In this section we prove Lemma 7, which is used for the proof of Theorem 6. To make our inductive proof of Lemma 7 work, we show in fact something more general, which requires the following definition. Given a graph G and a subset $F \subseteq V(G)$, a subset $S \subseteq V(G)$ is *F-avoiding* if $S \cap F = \varnothing$. By extension, a family $\mathcal{S} \subseteq 2^{V(G)}$ is *F-avoiding* if each member of $\mathcal{S}$ is F-avoiding. For a graph G, a family $\mathcal{S} \subseteq 2^{V(G)}$ of subsets of $V(G)$ is an *F-avoiding distance-d amiable family* if it satisfies the following properties:

- $\mathcal{S}$ is F-avoiding;
- Each member of $\mathcal{S}$ induces a P_3-free subgraph in G;
- For each $S \in \mathcal{S}$, the connected components of $G[S]$ are pairwise at distance at least d in G;
- Each (inclusion-wise) maximal F-avoiding distance-d independent set of G is contained in some member of $\mathcal{S}$.

Note that a distance-d amiable family of G is nothing but an F-avoiding distance-d amiable family of G for $F = \varnothing$. As we shall see, our proof of Lemma 7 in fact shows that, for every kP_3-free graph G and every $F \subseteq V(G)$, the graph G admits an F-avoiding distance-d amiable family of size $|V(G)|^{O(k)}$ and which can be computed in time $|V(G)|^{O(k)}$.

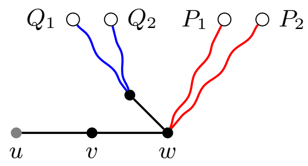

Figure 2 The case $d \leq 5$ (paths in blue are of length $d-3$, those in red, $d-2$).

Before formally proving Lemma 7, let us discuss why our approach fails for $3 \leq d \leq 5$ (recall however that the failure for $d \in \{3,5\}$ is to be expected given the hardness results in Theorem 2). The algorithm for computing a distance-d amiable family is, in essence, similar to the one for computing an amiable family (so $d = 2$): it enumerates all P_3's uvw in the graph, "guesses" which vertex in the path is in the independent set and recursively calls the algorithm on (roughly) the anti-neighborhood of $\{u, v, w\}$, which is $(k-1)P_3$-free. The main difference (and difficulty) is in ensuring that the family computed recursively satisfies the distance requirement: the connected components of each member in this family could in principle be closer in the original graph. Consider, for instance, the case where u is picked in the independent set, and there are four connected components Q_1, Q_2, P_1, P_2 in a member of the recursively computed family with shortest paths between Q_1 and Q_2, and P_1 and P_2 as shown in Figure 2. Then, for $d \leq 5$, Q_1 and Q_2 can end up at a distance less than d, while P_1 and P_2 are at a distance at least d. Since there is a priori no way of "distinguishing" these two cases, we have to take d "large enough". The precise argument appears in the proof of Lemma 7 (Claim 12). We finally restate Lemma 7.

▶ **Lemma 7.** *Let $d \geq 6$ be a fixed integer. For every $k \in \mathbb{N}$, every kP_3-free graph admits a distance-d amiable family of size $|V(G)|^{O(k)}$, which can be computed in time $|V(G)|^{O(k)}$.*

Proof of Lemma 7. To prove the lemma, we provide, for every $d \geq 6$ and every $k \geq 1$, an algorithm Λ_k^d which takes as input a kP_3-free graph G, together with an arbitrary ordering of $V(G)$, and a subset $F \subseteq V(G)$, and outputs an F-avoiding distance-d amiable family $\Lambda_k^d(G, F)$ of G. The pseudo-code of the algorithm is given in Algorithm 1. Note that, if the input graph G is the empty graph, Λ_k^d correctly outputs $\Lambda_k^d(G) = \{\varnothing\}$.

In the following, we prove three claims (Claims 11–13) which altogether show that, for every $d \geq 6$ and every $k \geq 1$, if G is a kP_3-free graph and $F \subseteq V(G)$, then the family $\Lambda_k^d(G, F)$ is indeed an F-avoiding distance-d amiable family. We will then show that $\Lambda_k^d(G, F)$ has size $|V(G)|^{O(k)}$ and that the algorithm Λ_k^d has running time $|V(G)|^{O(k)}$. Taking $F = \varnothing$, will prove the lemma.

Given a graph G on n vertices and an ordering $v_1, \ldots, v_n$ of $V(G)$, we let $G_i = G[\{v_1, \ldots, v_i\}]$. Moreover, an induced path in G with vertex set $\{x_1, x_2, \ldots, x_\ell\}$ and edge set $\{x_1x_2, x_2x_3, \ldots x_{\ell-1}x_\ell\}$ is denoted by listing its vertices in the natural order $x_1x_2 \cdots x_\ell$.

▷ Claim 11 (⋆). For every $d \geq 6$ and every $k \geq 1$, if G is a kP_3-free graph and $F \subseteq V(G)$, then $\Lambda_k^d(G, F)$ is F-avoiding.

▷ Claim 12. For every $d \geq 6$ and $k \geq 1$, if G is a kP_3-free graph, $F \subseteq V(G)$, and $S \in \Lambda_k^d(G, F)$, then $G[S]$ is P_3-free and its connected components are pairwise at distance at least d in G.

Proof of Claim 12. For fixed $d \geq 6$, we proceed by induction on k. The statement holds for $k = 1$, since for every P_3-free graph G and every $F \subseteq V(G)$, $\Lambda_1^d(G) = \{V(G) \setminus F\}$. Suppose now that $k > 1$ and that the statement holds for $k - 1$. Let G be an n-vertex kP_3-free

Algorithm 1 Λ_k^d.

Input: A kP_3-free graph G, an arbitrary ordering $v_1, \ldots, v_n$ of $V(G)$, and a subset $F \subseteq V(G)$.
Output: An F-avoiding distance-d amiable family of G.

1: Initialize $\mathcal{S} = \{\varnothing\}$.
2: **for** $i = 1, \ldots, n$ **do**
3:   **if** $v_i \notin F$ **then**
4:     **for** every member $S \in \mathcal{S}$ **do**
5:       **if** $G[S \cup \{v_i\}]$ is P_3-free and its connected components are pairwise at distance at least d in G **then**
6:         Set $S = S \cup \{v_i\}$.
7:     **for** every induced P_3 uvw in G_i such that $u \notin F$ **do**
8:       Compute $\mathcal{C} := \Lambda_{k-1}^d(G[N_G^{\geq 4}(u)], (F \cap N_G^{\geq 4}(u)) \cup (N_G^{\geq 4}(u) \cap N_G^{\leq d-1}(u)))$.
9:       **for** every $C \in \mathcal{C}$ **do**
10:         Set $\mathcal{S} = \mathcal{S} \cup \{C \cup \{u\}\}$.
11:     **for** every induced P_3 uvw in G_i such that $v \notin F$ **do**
12:       Compute $\mathcal{C} := \Lambda_{k-1}^d(G[N_G^{\geq 4}(v)], (F \cap N_G^{\geq 4}(v)) \cup (N_G^{\geq 4}(v) \cap N_G^{\leq d-1}(v)))$.
13:       **for** every $C \in \mathcal{C}$ **do**
14:         Set $\mathcal{S} = \mathcal{S} \cup \{C \cup \{v\}\}$.
15: **return** $\Lambda_k^d(G, F) := \mathcal{S}$.

graph and let $F \subseteq V(G)$ as in input. For every $i \in [n]$, let $\mathcal{S}^i$ be the state of the family $\mathcal{S}$ at the end of the i-th iteration of the main loop. We show by induction on i that each member of $\mathcal{S}^i$ induces a P_3-free subgraph of G whose connected components are pairwise at distance at least d in G. The case $i = 1$ trivially holds, since either $v_1 \in F$ in which case $\mathcal{S}^1 = \{\varnothing\}$, or $v_1 \notin F$ in which case $\mathcal{S}^1 = \{\{v_1\}\}$. Thus, suppose that $i > 1$ and that the statement holds for $i - 1$.

Consider $S \in \mathcal{S}^i$. If $S \in \mathcal{S}^{i-1}$ then, by the induction hypothesis, $G[S]$ is P_3-free and its connected components are pairwise at distance at least d in G. Thus, we may assume that $S \notin \mathcal{S}^{i-1}$. This implies that S is added to $\mathcal{S}^i$ in one of the three inner loops during the i-th iteration of the main loop. If S is added in line 6 as an extension of a member of $\mathcal{S}^{i-1}$, then the statement holds by construction. Suppose next that S is added to $\mathcal{S}^i$ in line 10. Hence, there exist an induced P_3 uvw in G_i such that $u \notin F$ and a set $C \in \Lambda_{k-1}^d(G[N_G^{\geq 4}(u)], (F \cap N_G^{\geq 4}(u)) \cup (N_G^{\geq 4}(u) \cap N_G^{\leq d-1}(u)))$ such that $S = C \cup \{u\}$. Observe now that since G is kP_3-free and $(N_G(v) \cup N_G(w)) \cap N_G^{\geq 4}(u) = \varnothing$, the graph $G[N_G^{\geq 4}(u)]$ is $(k-1)P_3$-free, and thus, by the induction hypothesis on $k - 1$, every member of $\Lambda_{k-1}^d(G[N_G^{\geq 4}(u)], (F \cap N_G^{\geq 4}(u)) \cup (N_G^{\geq 4}(u) \cap N_G^{\leq d-1}(u)))$ induces a P_3-free subgraph of G whose connected components are pairwise at distance at least d in $G[N_G^{\geq 4}(u)]$. We claim that then, the connected components of $G[C]$ are in fact pairwise at distance at least d in G. Suppose, to the contrary, that there exist two connected components Q_1 and Q_2 of $G[C]$ such that $d_G(Q_1, Q_2) \leq d - 1$. Since the distance in $G[N_G^{\geq 4}(u)]$ between Q_1 and Q_2 is at least d, every shortest path in G from Q_1 to Q_2 contains at least one vertex of $V(G) \setminus N_G^{\geq 4}(u)$; let P be such a shortest path and let $x \in V(G) \setminus N_G^{\geq 4}(u)$ be an arbitrary vertex on P. Clearly, $d_G(u, x) \leq 3$. Now, by Claim 11, $C \cap ((F \cap N_G^{\geq 4}(u)) \cup (N_G^{\geq 4}(u) \cap N_G^{\leq d-1}(u))) = \varnothing$ which implies, in particular, that $C \subseteq N_G^{\geq d}(u)$. It follows that for $j \in [2]$,

$$d \leq d_G(u, Q_j) \leq d_G(u, x) + d_G(x, Q_j) \leq 3 + d_G(x, Q_j) \text{ and thus,}$$
$$d_G(Q_1, Q_2) = d_G(Q_1, x) + d_G(x, Q_2) \geq 2(d - 3).$$

Since, by assumption, $d_G(Q_1, Q_2) \leq d-1$, we conclude that $2(d-3) \leq d-1$, that is, $d \leq 5$, a contradiction. Thus, the connected components of $G[C]$ are pairwise at distance at least d in G. Since $C \subseteq N_G^{\geq d}(u)$, as previously observed, and $G[C]$ is P_3-free, it follows that $S = C \cup \{u\}$ induces a P_3-free graph whose connected components are pairwise at distance at least d in G. We conclude similarly in the case where S is added to $\mathcal{S}^i$ in line 14. ◁

▷ Claim 13. For every $d \geq 6$ and $k \geq 1$, if G is a kP_3-free graph and $F \subseteq V(G)$, then every F-avoiding distance-d independent set of G is contained in a member of $\Lambda_k^d(G, F)$.

Proof of Claim 13. For fixed $d \geq 6$, we proceed by induction on k. The statement clearly holds for $k = 1$, since for every P_3-free graph G and every $F \subseteq V(G)$, $\Lambda_1^d(G, F) = \{V(G) \setminus F\}$. Suppose now that $k > 1$ and that the statement holds for $k-1$. Let G be a kP_3-free graph and let $F \subseteq V(G)$ as in input. For every $i \in [n]$, let $\mathcal{S}^i$ be the state of the family $\mathcal{S}$ at the end of the i-th iteration of the main loop. Furthermore, given a distance-d independent set I of G such that $I \subseteq V(G_i)$, we say that I is *G_i-compatible* if for every $u \in I$ and every connected component Q of G_i such that u is connected to Q in G, there exists a shortest u, Q-path $P_{u,Q}$ in G such that $N_G^{\leq 2}(u) \cap V(P_{u,Q}) \subseteq V(G_i)$. We now show, by induction on i, that for every G_i-compatible F-avoiding distance-d independent set I of G such that $I \subseteq V(G_i)$, there exists $S \in \mathcal{S}^i$ such that $I \subseteq S$. Note that this is enough to prove our statement for k, since any F-avoiding distance-d independent set of $G = G_n$ is surely G_n-compatible.

The base case $i = 1$ trivially holds, since either $v_1 \in F$ and $\mathcal{S}^1 = \{\varnothing\}$ in which case $\varnothing$ is the only G_1-compatible F-avoiding distance-d independent of G_1, or $v_1 \notin F$ and $\mathcal{S}^1 = \{\{v_1\}\}$ in which case $\{v_1\}$ is the only maximal G_1-compatible F-avoiding distance-d independent set of G_1. Thus, suppose that $i > 1$ and that the statement holds for $i-1$.

Consider a G_i-compatible F-avoiding distance-d independent set I of G such that $I \subseteq V(G_i)$. If $I \subseteq V(G_{i-1})$ and I is G_{i-1}-compatible then, by the induction hypothesis, there exists $S \in \mathcal{S}^{i-1}$ such that $I \subseteq S$. Observe now that, by construction, there then exists $S' \in \mathcal{S}^i$ such that $S \subseteq S'$ and hence $I \subseteq S'$. Thus, assume that this does not hold. This implies that either $I \not\subseteq V(G_{i-1})$, or $I \subseteq V(G_{i-1})$ but I is not G_{i-1}-compatible. We distinguish these two cases and argue that we can always find $S \in \mathcal{S}^i$ such that $I \subseteq S$.

Case 1. $I \not\subseteq V(G_{i-1})$.
Hence, $v_i \in I$. Since I is F-avoiding, $v_i \notin F$. We claim that $I \setminus \{v_i\}$ is G_{i-1}-compatible. Indeed, since I is G_i-compatible, for every $u \in I \setminus \{v_i\}$ and every connected component Q of G_i such that u is connected to Q in G, there exists a shortest u, Q-path $P_{u,Q}$ in G such that $N_G^{\leq 2}(u) \cap V(P_{u,Q}) \subseteq V(G_i)$. But v_i is at distance at least $d \geq 6$ from u in G and so $v_i \notin (N_G^{\leq 2}(u) \cap V(P_{u,Q}))$, that is, $N_G^{\leq 2}(u) \cap V(P_{u,Q}) \subseteq V(G_{i-1})$ which proves our claim. By the induction hypothesis, there exists $S \in \mathcal{S}^{i-1}$ such that $I \setminus \{v_i\} \subseteq S$. If $G[S \cup \{v_i\}]$ is P_3-free and its connected components are pairwise at distance at least d in G, then $S \cup \{v_i\} \in \mathcal{S}^i$ by construction (cf. line 6) and $I \subseteq S \cup \{v_i\}$. Thus, we may assume that this does not hold. Hence, either $G[S \cup \{v_i\}]$ contains at least one induced P_3, or $G[S \cup \{v_i\}]$ is P_3-free but at least two of its connected components are at distance strictly less than d in G.

Case 1.1. $G[S \cup \{v_i\}]$ contains at least one induced P_3.
By Claim 12, $G[S]$ is P_3-free and thus, any induced P_3 in $G[S \cup \{v_i\}]$ must contain v_i. We distinguish two cases. Suppose first that there exist a connected component Q of $G[S]$ and a vertex $u \in Q$ such that v_i is adjacent to u but not complete to Q, say $w \in Q$ is nonadjacent to v_i. Note that at some point during the i-th iteration of the main loop, the induced P_3 $v_i uw$ is considered in the second inner loop, since $v_i \notin F$. Furthermore, since G is kP_3-free and $(N_G(u) \cup N_G(w)) \cap N_G^{\geq 4}(v_i) = \varnothing$, the graph $G[N_G^{\geq 4}(v_i)]$ is $(k-1)P_3$-free. Thus, by the induction hypothesis on $k-1$ and since $I \setminus \{v_i\} \subseteq N_G^{\geq d}(v_i) \setminus F$, there exists

$C \in \Lambda^d_{k-1}(G[N^{\geq 4}_G(v_i)], (F \cap N^{\geq 4}_G(v_i)) \cup (N^{\geq 4}_G(v_i) \cap N^{\leq d-1}_G(v_i)))$ such that $I \setminus \{v_i\} \subseteq C$. But then, $C \cup \{v_i\} \in \mathcal{S}^i$ and $I \subseteq C \cup \{v_i\}$. The case where, for every connected component Q of $G[S]$ the vertex v_i is either complete or anticomplete to Q, is handled similarly.

Case 1.2. $G[S \cup \{v_i\}]$ is P_3-free but at least two of its connected components are at distance strictly less than d in G.
By Claim 12, the connected components of $G[S]$ are pairwise at distance at least d in G and so there must exist a connected component Q of $G[S \cup \{v_i\}]$ at distance strictly less than d in G to the connected component of $G[S \cup \{v_i\}]$ containing v_i. Now, since v_i is connected to Q in G and $v_i \in I$, the assumption that I is G_i-compatible implies that there exists a shortest v_i, Q-path $P_{v_i,Q}$ in G such that $N^{\leq 2}_G(v_i) \cap V(P_{v_i,Q}) \subseteq V(G_i)$. Let $u_1, u_2 \in V(G_i)$ be the two vertices on $P_{v_i,Q}$ at distance one and two, respectively, from v_i. Note that at some point during the i-th iteration of the main loop, the induced P_3 $v_i u_1 u_2$ is considered in the second inner loop, since $v_i \notin F$. Moreover, since G is kP_3-free and $(N_G(u_1) \cup N_G(u_2)) \cap N^{\geq 4}_G(v_i) = \varnothing$, the graph $G[N^{\geq 4}_G(v_i)]$ is $(k-1)P_3$-free. Thus, by the induction hypothesis on $k-1$ and since $I \subseteq N^{\geq d}_G(v_i) \setminus F$, there exists $C \in \Lambda^d_{k-1}(G[N^{\geq 4}_G(v_i)], (F \cap N^{\geq 4}_G(v_i)) \cup (N^{\geq 4}_G(v_i) \cap N^{\leq d-1}_G(v_i)))$ such that $I \setminus \{v_i\} \subseteq C$. But then, $C \cup \{v_i\} \in \mathcal{S}^i$ and $I \subseteq C \cup \{v_i\}$.

Case 2. $I \subseteq V(G_{i-1})$ but I is not G_{i-1}-compatible.
Hence, there must exist $u \in I$ and a connected component Q of G_i to which u is connected in G such that $v_i \in N^{\leq 2}_G(u) \cap V(P_{u,Q})$, where $P_{u,Q}$ is a shortest path in G from u to Q given by the G_i-compatibility of I. Let $u_1, u_2 \in V(G_i)$ be the vertices on $P_{u,Q}$ at distance one and two, respectively, from u. Note that at some point during the i-th iteration of the main loop, the induced P_3 $u u_1 u_2$ is considered in the second inner loop, since $v_i \notin F$. Moreover, since G is kP_3-free and $(N_G(u_1) \cup N_G(u_2)) \cap N^{\geq 4}_G(u) = \varnothing$, the graph $G[N^{\geq 4}_G(u)]$ is $(k-1)P_3$-free. Thus, by the induction hypothesis on $k-1$ and since $I \subseteq N^{\geq d}_G(u) \setminus F$, there exists $C \in \Lambda^d_{k-1}(G[N^{\geq 4}_G(u)], (F \cap N^{\geq 4}_G(u)) \cup (N^{\geq 4}_G(u) \cap N^{\leq d-1}_G(u)))$ such that $I \setminus \{u\} \subseteq C$. But then, $C \cup \{u\} \in \mathcal{S}^i$ and $I \subseteq C \cup \{u\}$.

Since in all cases we found $S \in \mathcal{S}^i$ with $I \subseteq S$, this concludes the proof of Claim 13. ◁

It follows now from Claims 11–13 that, for every $d \geq 6$ and every $k \geq 1$, if G is a kP_3-free graph and $F \subseteq V(G)$, then the family $\Lambda^d_k(G, F)$ is indeed an F-avoiding distance-d amiable family. It remains to show that $\Lambda^d_k(G, F)$ has size at most $|V(G)|^{O(k)}$ and that the running time of the algorithm Λ^d_k is $|V(G)|^{O(k)}$. To this end, let

$$f_d(n, k) = \max\{|\Lambda^d_k(G, F)| : |V(G)| \leq n,\ F \subseteq V(G) \text{ and } G \text{ is } kP_3\text{-free}\}.$$

Clearly, $f_d(n, 1) = 1$ for every $n \in \mathbb{N}$. We claim that, for every $n \in \mathbb{N}$ and $k > 1$, $f_d(n, k) \leq 2n^4 \cdot f_d(n, k-1)$. Indeed, for every n-vertex kP_3-free graph G and every $F \subseteq V(G)$, a member of $\Lambda^d_k(G, F)$ can only be created during an i-th iteration of the main loop (for some $i \in [n]$) in one of the inner loops from an induced P_3 of G_i and some set resulting from a call to Λ^d_{k-1}. Since for each $i \in [n]$ there are at most i^3 such copies of P_3 in G_i, at most $2i^3 \cdot f_d(n, k-1)$ new members are added in the i-th iteration of the main loop. It follows that $f_d(n, k) \leq \sum_{i \in [n]} 2i^3 \cdot f_d(n, k-1) \leq 2n^4 \cdot f_d(n, k-1)$ and thus $f_d(n, k) \leq 2^{k-1} n^{4(k-1)}$.

Similarly, for every $d \geq 6$ and every $n, k \in \mathbb{N}$, if $T_d(n, k)$ denotes the running time of the algorithm Λ^d_k on an n-vertex kP_3-free graph, then clearly $T_d(n, 1) = O(n)$. Furthermore, we obtain the following recurrence for $T_d(n, k)$, where we use the fact that checking if an n-vertex graph is P_3-free can be done in $O(n^3)$ time and determining if the connected components of an n-vertex graph are pairwise at distance at least d can be done in $O(n^3)$ time (using an all-pair-shortest-path algorithm):

$$T_d(n, k) \leq cn \cdot (f_d(n, k) \cdot n^3 + 2n^3 \cdot (T_d(n, k-1) + f_d(n, k-1))) \leq 2cn^4 \cdot T_d(n, k-1) + O(n^{4k}),$$

for some constant $c > 0$. We conclude that $T_d(n, k) \leq n^{O(k)}$. ◀

References

1 Akanksha Agrawal, Paloma T Lima, Daniel Lokshtanov, Paweł Rzążewski, Saket Saurabh, and Roohani Sharma. Odd Cycle Transversal on P_5-free graphs in polynomial time. *ACM Transactions on Algorithms*, 21(2):16:1–16:14, 2025.

2 Gábor Bacsó, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Zsolt Tuza, and Erik Jan van Leeuwen. Subexponential-time algorithms for Maximum Independent Set in P_t-free and broom-free graphs. *Algorithmica*, 81(2):421–438, 2019.

3 Nina Chiarelli, Tatiana R Hartinger, Matthew Johnson, Martin Milanič, and Daniël Paulusma. Minimum connected transversals in graphs: new hardness results and tractable cases using the price of connectivity. *Theoretical Computer Science*, 705:75–83, 2018. doi:10.1016/J.TCS.2017.09.033.

4 Maria Chudnovsky, Sepehr Hajebi, and Sophie Spirkl. List-k-Coloring H-free graphs for all $k > 4$. *Combinatorica*, 44(5):1063–1068, 2024.

5 Maria Chudnovsky, Rose McCarty, Marcin Pilipczuk, Michał Pilipczuk, and Paweł Rzążewski. Sparse induced subgraphs in P_6-free graphs. In David P. Woodruff, editor, *Proceedings of the 2024 ACM-SIAM Symposium on Discrete Algorithms (SODA 2024)*, pages 5291–5299. SIAM, 2024. doi:10.1137/1.9781611977912.190.

6 Jean-François Couturier, Petr A Golovach, Dieter Kratsch, and Daniël Paulusma. List coloring in the absence of a linear forest. *Algorithmica*, 71:21–35, 2015. doi:10.1007/S00453-013-9777-0.

7 Konrad K Dabrowski, Carl Feghali, Matthew Johnson, Giacomo Paesani, Daniël Paulusma, and Paweł Rzążewski. On cycle transversals and their connected variants in the absence of a small linear forest. *Algorithmica*, 82:2841–2866, 2020. doi:10.1007/S00453-020-00706-6.

8 Hiroshi Eto, Fengrui Guo, and Eiji Miyano. Distance-d independent set problems for bipartite and chordal graphs. *Journal of Combinatorial Optimization*, 27(1):88–99, 2014.

9 Sepehr Hajebi, Yanjia Li, and Sophie Spirkl. Complexity dichotomy for List-5-Coloring with a forbidden induced subgraph. *SIAM Journal on Discrete Mathematics*, 36(3):2004–2027, 2022. doi:10.1137/21M1443352.

10 Cicely Henderson, Evelyne Smith-Roberge, Sophie Spirkl, and Rebecca Whitman. Maximum k-colourable induced subgraphs in $(P_5 + rK_1)$-free graphs. *arXiv*, 2024. arXiv:2410.08077.

11 Richard M. Karp. Reducibility among combinatorial problems. In Raymond E. Miller, James W. Thatcher, and Jean D. Bohlinger, editors, *Complexity of Computer Computations: Proceedings of a symposium on the Complexity of Computer Computations*, pages 85–103. Springer US, 1972. doi:10.1007/978-1-4684-2001-2_9.

12 John M. Lewis and Mihalis Yannakakis. The node-deletion problem for hereditary properties is NP-complete. *Journal of Computer and System Sciences*, 20(2):219–230, 1980. doi:10.1016/0022-0000(80)90060-4.

13 Daniel Lokshtanov, Paweł Rzążewski, Saket Saurabh, Roohani Sharma, and Meirav Zehavi. Maximum Partial List H-Coloring on P_5-free graphs in polynomial time. *arXiv*, 2024. arXiv:2410.21569.

14 Vadim V. Lozin. From matchings to independent sets. *Discrete Applied Mathematics*, 231:4–14, 2017. doi:10.1016/J.DAM.2016.04.012.

15 Vadim V. Lozin and Raffaele Mosca. Maximum regular induced subgraphs in $2P_3$-free graphs. *Theoretical Computer Science*, 460:26–33, 2012.

16 Giacomo Paesani, Daniël Paulusma, and Paweł Rzążewski. Feedback Vertex Set and Even Cycle Transversal for H-free graphs: Finding large block graphs. *SIAM Journal on Discrete Mathematics*, 36(4):2453–2472, 2022. doi:10.1137/22M1468864.

17 Alexander Schrijver. *Combinatorial Optimization – Polyhedra and Efficiency*, volume 24 of *Algorithms and Combinatorics*. Springer, 2004.

18 Alexa Sharp. Distance coloring. In Lars Arge, Michael Hoffmann, and Emo Welzl, editors, *Algorithms – ESA 2007, 15th Annual European Symposium*, volume 4698 of *Lecture Notes in Computer Science*, pages 510–521. Springer, 2007. doi:10.1007/978-3-540-75520-3_46.