Colouring Graphs of Bounded Diameter in the Absence of Small Cycles*

Barnaby Martin[1], Daniël Paulusma[2][0000−0001−5945−9287], and Siani Smith[1]

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

Abstract. For $k \geq 1$, a k-colouring c of G is a mapping from $V(G)$ to $\{1, 2, \ldots, k\}$ such that $c(u) \neq c(v)$ for any two non-adjacent vertices u and v. The k-Colouring problem is to decide if a graph G has a k-colouring. For a family of graphs $\mathcal{H}$, a graph G is $\mathcal{H}$-free if G does not contain any graph from $\mathcal{H}$ as an induced subgraph. Let C_s be the s-vertex cycle. In previous work (MFCS 2019) we examined the effect of bounding the diameter on the complexity of 3-Colouring for $(C_3, \ldots, C_s)$-free graphs and H-free graphs where H is some polyad. Here, we prove for certain small values of s that 3-Colouring is polynomial-time solvable for C_s-free graphs of diameter 2 and (C_4, C_s)-free graphs of diameter 2. In fact, our results hold for the more general problem List 3-Colouring. We complement these results with some hardness result for diameter 4.

1 Introduction

Graph colouring is a well-studied topic in Computer Science due to its wide range of applications. A *k-colouring* of a graph G is a mapping $c : V(G) \to \{1, \ldots, k\}$ that assigns each vertex u a *colour* $c(u)$ in such a way that $c(u) \neq c(v)$ for any two adjacent vertices u and v of G. The aim is to find the smallest value of k (also called the *chromatic number*) such that G has a k-colouring. The corresponding decision problem is called Colouring, or k-Colouring if k is fixed, that is, not part of the input. As even 3-Colouring is NP-complete [16], k-Colouring and Colouring have been studied for many special graph classes, as surveyed in, for example, [1,5,9,13,15,21,23,26]. This holds in particular for *hereditary* classes of graphs, which are the classes of graphs closed under vertex deletion.

It is well known and not difficult to see that a class of graphs is hereditary if and only if it can be characterized by a unique set $\mathcal{F}_{\mathcal{G}}$ of minimal forbidden induced subgraphs. In particular, a graph G is *H-free* for some graph H if G does not contain H as an *induced* subgraph. The latter means that we cannot modify G into H by a sequence of vertex deletions. For a set of graphs $\{H_1, \ldots, H_p\}$, a graph G is *$(H_1, \ldots, H_p)$-free* if G is H_i-free for every $i \in \{1, \ldots, p\}$.

We continue a long-term study on the complexity of 3-Colouring for special graph classes. Let C_t and P_t be the cycle and path, respectively, on t vertices. The complexity of 3-Colouring for H-free graphs has not yet been classified; in

* Research supported by the Leverhulme Trust (RPG-2016-258).

particular this is still is open for P_t-free graphs for every $t \geq 8$, whereas the case $t = 7$ is polynomial [3]. For $t \geq 3$, let $C_{>t} = \{C_{t+1}, C_{t+2}, \ldots\}$. Note that for $t \geq 2$, the class of P_t-free graphs is a subclass of $C_{>t}$-free graphs. Recently, Pilipczuk, Pilipczuk and Rzążewski [22] gave for every $t \geq 3$, a quasi-polynomial-time algorithm for 3-Colouring on $C_{>t}$-free graphs. Rojas and Stein [24] proved in another recent paper that for every odd integer $t \geq 9$, 3-Colouring is polynomial-time solvable for $(\mathcal{C}^{odd}_{<t-3}, P_t)$-free graphs, where $\mathcal{C}^{odd}_{<t}$ is the set of all odd cycles on less than t vertices. This complements a result from [10], which implies that for every $t \geq 1$, 3-Colouring, or more general List 3-Colouring (defined later), is polynomial-time solvable for (C_4, P_t)-free graphs (see also [18]).

The graph classes in this paper are only partially characterized by forbidden induced subgraphs: we also restrict the diameter. The *distance* $\text{dist}(u, v)$ between two vertices u and v in a graph G is the length (number of edges) of a shortest path between them. The *diameter* of a graph G is the maximum distance over all pairs of vertices in G. Note that the n-vertex path P_n has diameter $n - 1$, but by removing an internal vertex the diameter becomes infinite. Hence, for every integer $d \geq 2$, the class of graphs of diameter at most d is not hereditary.

For every $d \geq 3$, the 3-Colouring problem for graphs of diameter at most d is NP-complete, as shown by Mertzios and Spirakis [20] who gave a highly non-trivial NP-hardness construction for the case where $d = 3$. In fact they proved that 3-Colouring is NP-complete even for C_3-free graphs of diameter 3 and radius 2. The complexity of 3-Colouring for the class of all graphs of diameter 2 has been posed as an open problem in several papers [2,4,19,20,21].

On the positive side, Mertzios and Spirakis [20] gave a subexponential-time algorithm for 3-Colouring on graphs of diameter 2. Moreover, as we discuss below, 3-Colouring is polynomial-time solvable for several subclasses of diameter 2. A graph G has an *articulation neighbourhood* if $G - (N(v) \cup \{v\})$ is disconnected for some $v \in V(G)$. The neighbourhoods $N(u)$ and $N(v)$ of two distinct (and non-adjacent) vertices u and v are *nested* if $N(u) \subseteq N(v)$. We let $K_{1,r}$ be the star on $r + 1$ vertices. The *subdivision* of an edge uw in a graph removes uw and replaces it with a new vertex v and edges uv, vw. We let $K^{\ell}_{1,r}$ be the *ℓ-subdivided star*, which is obtained from $K_{1,r}$ by subdividing *one* edge exactly ℓ times. The graph $S_{h,i,j}$, for $1 \leq h \leq i \leq j$, is the tree with one vertex x of degree 3 and exactly three leaves, which are of distance h, i and j from x, respectively. Note that $S_{1,1,1} = K_{1,3}$. The *diamond* is obtained from the 4-vertex complete graph by deleting an edge. The 3-Colouring problem is polynomial-time solvable for:

- diamond-free graphs of diameter 2 with an articulation neighbourhood but without nested neighbourhoods [20];
- (C_3, C_4)-free graphs of diameter 2 [19];
- $K^2_{1,r}$-free graphs of diameter 2, for every $r \geq 1$ [19]; and
- $S_{1,2,2}$-free graphs of diameter 2 [19].

It follows from results in [8,12,17] that without the diameter-2 condition, 3-Colouring is NP-complete again in each of the above cases; in particular 3-Colouring is NP-complete for $\mathcal{C}$-free graphs for any finite set $\mathcal{C}$ of cycles.

Our Results. We aim to increase our understanding of the complexity of 3-Colouring for graphs of diameter 2. In [19] we mainly considered 3-Colouring for graphs of diameter 2 with some forbidden induced subdivided star. In this paper, we continue this study by focussing on 3-Colouring for C_s-free or (C_s, C_t)-free graphs of diameter 2 for small values of s and t; in particular for the case where $s = 4$ (cf. the aforementioned result for (C_4, P_t)-free graphs). In fact we prove our results for a more general problem, namely List 3-Colouring, whose complexity for diameter 2 is also still open. A *list assignment* of a graph $G = (V, E)$ is a function L that prescribes a *list of admissible colours* $L(u) \subseteq \{1, 2, \ldots\}$ to each $u \in V$. A colouring c *respects* L if $c(u) \in L(u)$ for every $u \in V$. For an integer $k \geq 1$, if $L(u) \subseteq \{1, \ldots, k\}$ for each $u \in V$, then L is a *list k-assignment*. The List k-Colouring problem is to decide if a graph G with an list k-assignment L has a colouring that respects L. If every list is $\{1, \ldots, k\}$, we obtain k-Colouring.

The following two theorems summarize our main results.

Theorem 1. *For $s \in \{5, 6\}$,* List 3-Colouring *is polynomial-time solvable for C_s-free graphs of diameter* 2.

Theorem 2. *For $t \in \{3, 5, 6, 7, 8, 9\}$,* List 3-Colouring *is polynomial-time solvable for (C_4, C_t)-free graphs of diameter* 2.

The case $t = 3$ in Theorem 2 directly follows from the Hoffman-Singleton Theorem [11], which states that there are only four (C_3, C_4)-free graphs of diameter 2. The cases $t \in \{5, 6\}$ immediately follows from Theorem 1. Hence, apart from proving Theorem 1, we only need to prove Theorem 2 for $t \in \{7, 8, 9\}$.

We prove Theorem 1 and the case $t = 7$ of Theorem 2 in Section 3. As we explain in the same section, all these results follow from the same technique, which is based on a number of (known) propagation rules. We first colour a small number of vertices and then start to apply the propagation rules exhaustively. This will reduce the sizes of the lists of the vertices. The novelty of our approach is the following: we can prove that the diameter-2 property ensures such a widespread reduction that each precolouring changes our instance into an instance of 2-List Colouring: the polynomial-solvable variant of List Colouring where each list has size at most 2 [7] (see also Section 2).

We prove the cases $t = 8$ and $t = 9$ of Theorem 2 in Section 4 using a refinement of the technique from Section 3. We explain this refinement in detail at the start of Section 4. In short, in our branching, we exploit information from earlier obtained no-answers to reduced instances of our original instance (G, L).

We complement Theorems 1 and 2 by the following result for diameter 4, whose proof we omit.

Theorem 3. *For every even integer $t \geq 6$,* 3-Colouring *is* NP*-complete on the class of $(C_4, C_6, \ldots, C_t)$-free graphs of diameter* 4.

Results of Damerell [6] imply that 3-Colouring is polynomial-time solvable for (C_3, C_4, C_5, C_6)-free graphs of diameter 3 and for $(C_3, \ldots, C_8)$-free graphs of diameter 4 [19]. We were not able to reduce the diameter in Theorem 3 from 4 to 3; see Section 5 for a further discussion, including other open problems.

2 Preliminaries

Let $G = (V, E)$ be a graph. A vertex $u \in V$ is *dominating* if u is adjacent to every other vertex of G. For $S \subseteq V$, the graph $G[S] = (S, \{uv \mid u, v \in S \text{ and } uv \in E\})$ denotes the subgraph of G induced by S. The *neighbourhood* of a vertex $u \in V$ is the set $N(u) = \{v \mid uv \in E\}$ and the *degree* of u is the size of $N(u)$. For a set $U \subseteq V$, we write $N(U) = \bigcup_{u \in U} N(u) \setminus U$.

The *bull* is the graph obtained from a triangle on vertices x, y, z after adding two new vertices u and v and edges xu and yv. A *clique* is a set of pairwise adjacent vertices, and an *independent set* is a set of pairwise non-adjacent vertices.

Let G be a graph with a list assignment L. If $|L(u)| \leq \ell$ for each $u \in V$, then L is a *ℓ-list assignment.* A list k-assignment is a k-list assignment, but the reverse is not necessarily true. The ℓ-List Colouring problem is to decide if a graph G with an ℓ-list assignment L has a colouring that respects L. We use a known general strategy for obtaining a polynomial-time algorithm for List 3-Colouring on some class $\mathcal{G}$. That is, we will reduce the input to a polynomial number of instances of 2-List Colouring and use a well-known result:

Theorem 4 ([7]). *The* 2-List Colouring *problem is linear-time solvable.*

We also need an observation (proof omitted).

Lemma 1. *Let G be a non-bipartite graph of diameter 2. Then G contains a C_3 or induced C_5.*

3 The Propagation Algorithm and Three Results

We present our initial propagation algorithm, which is based on a number of (well-known) propagation rules; we illustrate Rules 4 and 5 in Figures 1 and 2.

Rule 1. **(no empty lists)** If $L(u) = \emptyset$ for some $u \in V$, then return `no`.
Rule 2. **(not only lists of size 2)** If $|L(u)| \leq 2$ for every $u \in V$, then apply Theorem 4.
Rule 3. **(single colour propagation)** If u and v are adjacent, $|L(u)| = 1$, and $L(u) \subseteq L(v)$, then set $L(v) := L(v) \setminus L(u)$.
Rule 4. **(diamond colour propagation)** If u and v are adjacent and share two common non-adjacent neighbours x and y with $|L(x)| = |L(y)| = 2$ and $L(x) \neq L(y)$, then set $L(x) := L(x) \cap L(y)$ and $L(y) := L(x) \cap L(y)$ (so $L(x)$ and $L(y)$ get size 1).
Rule 5. **(bull colour propagation)** If u and v are the two degree-1 vertices of an induced bull B of G and $L(u) = L(v) = \{i\}$ for some $i \in \{1, 2, 3\}$ and moreover $L(w) \neq \{i\}$ for the degree-2 vertex w of B, then set $L(w) := L(w) \cap \{i\}$.

We say that a propagation rule is *safe* if the new instance is a yes-instance of List 3-Colouring if and only if the original instance is so. We make the following observation, which is straightforward (see also [14]).

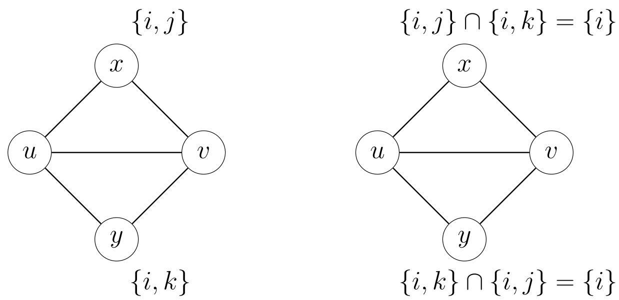

Fig. 1. Left: A diamond graph before applying Rule 4. Right: After applying Rule 4.

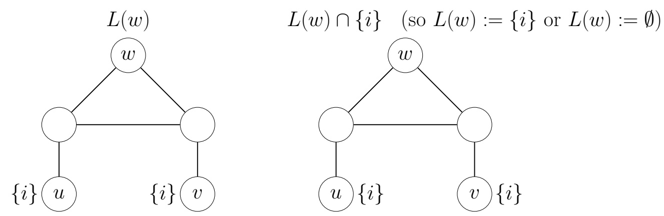

Fig. 2. Left: A bull graph before applying Rule 5. Right: After applying Rule 5.

Lemma 2. *Each of the Rules 1–5 is safe and can be applied in polynomial time.*

Consider again an instance (G, L). Let N_0 be a subset of $V(G)$ that has size at most some constant. Assume that $G[N_0]$ has a colouring c that respects the restriction of L to N_0. We say that c is an *L-promising N_0-precolouring* of G.

In our algorithms we first determine a set N_0 of constant size and consider every L-promising N_0-precolouring of G. That is, we modify L into a list assignment L_c with $L_c(u) = \{c(u)\}$ (where $c(u) \in L(u)$) for every $u \in N_0$ and $L_c(u) = L(u)$ for every $u \in V(G) \setminus N_0$). We then apply Rules 1–5 on (G, L_c) *exhaustively*, that is, until none of the rules can be applied anymore. This is the *propagation algorithm* and we say that it did a *full c-propagation*. The propagation algorithm may output `yes` and `no` (when applying Rules 1 or 2); else it will output `unknown`.

If the algorithm returns `yes`, then (G, L) is a yes-instance of List 3-Colouring by Lemma 2. If it returns `no`, then (G, L) has no L-respecting colouring coinciding with c on N_0, again by Lemma 2. If the algorithm returns `unknown`, then (G, L) may still have an L-respecting colouring that coincides with c on N_0. In that case the propagation algorithm did not apply Rule 1 or 2. Hence, it modified L_c into a list assignment L'_c of G such that $L'_c(u) \neq \emptyset$ for every $u \in V(G)$ and at least one vertex v of G still has a list $L'_c(v)$ of size 3, that is, $L'_c(v) = \{1, 2, 3\}$. We say that L'_c (if it exists) is the *c-propagated list assignment* of G.

After performing a full c-propagation for every L-promising N_0-precolouring c of G we say that we performed a *full N_0-propagation*. We say that (G, L) is *N_0-terminal* if after the full N_0-propagation one of the following cases hold:

1. for some L-promising N_0-precolouring, the propagation algorithm returned `yes`;
2. for every L-promising N_0-precolouring, the propagation algorithm returned `no`.

Note that if (G, L) is N_0-terminal for some set N_0, then we have solved List 3-Colouring on instance (G, L). The next lemma formalizes our approach (proof omitted).

Lemma 3. *Let (G, L) be an instance of* List 3-Colouring. *Let N_0 be a subset of $V(G)$ of constant size. Performing a full N_0-propagation takes polynomial time. Moreover, if (G, L) is N_0-terminal, then we have solved* List 3-Colouring *on instance (G, L).*

We now prove our first three results on List 3-Colouring for diameter-2 graphs. The first result, whose proof we omit, generalizes a corresponding result for 3-Colouring in [19].

Theorem 5. List 3-Colouring *can be solved in polynomial time for C_5-free graphs of diameter at most 2.*

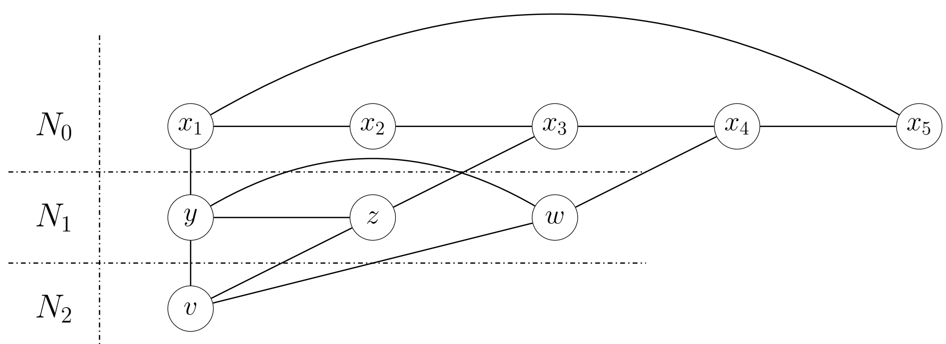

Fig. 3. The situation in the proof of Theorem 6, which is similar to the situation in the proof of Theorem 7.

Theorem 6. List 3-Colouring *can be solved in polynomial time for C_6-free graphs of diameter at most 2.*

Proof. Let $G = (V, E)$ be a C_6-free graph of diameter 2 with a list 3-assignment L. If G is C_5-free, then we apply Theorem 5. If G contains a K_4, then G is not 3-colourable and hence, (G, L) is a no-instance of List 3-Colouring. We check these properties in polynomial time. So, from now on, we assume that G is a K_4-free graph that contains an induced 5-vertex cycle C, say with vertex set $N_0 = \{x_1, \ldots, x_5\}$ in this order. Let N_1 be the set of vertices that do not belong to C but that are adjacent to at least one vertex of C. Let $N_2 = V \setminus (N_0 \cup N_1)$ be the set of remaining vertices.

As N_0 has size 5, we can apply a full N_0-propagation in polynomial time by Lemma 3. By the same lemma we are done if we can prove that (G, L) is N_0-terminal. We prove this claim below.

For contradiction, assume that (G, L) is not N_0-terminal. Then there must exist an L-promising N_0-precolouring c for which we obtain the c-propagated list assignment L'_c. By definition of L'_c we find that G contains a vertex v with $L'_c(v) = \{1, 2, 3\}$. Then $v \notin N_0$, as every $u \in N_0$ has $L'_c(u) = \{c(u)\}$. Moreover, $v \notin N_1$, as vertices in N_1 have a list of size at most 2 after applying Rule 3. Hence, we find that $v \in N_2$.

We first note that some colour of $\{1, 2, 3\}$ appears exactly once on N_0, as $|N_0| = 5$. Hence, we may assume without loss of generality that $c(x_1) = 1$ and that $c(x_i) \in \{2, 3\}$ for every $i \in \{2, 3, 4, 5\}$.

As G has diameter 2, there exists a vertex $y \in N_1$ that is adjacent to x_1 and v. As $L'_c(v) = \{1, 2, 3\}$ and $c(x_1) = 1$, we find that $L'_c(y) = \{2, 3\}$. As $c(x_i) \in \{2, 3\}$ for every $i \in \{2, 3, 4, 5\}$, the latter means that y is not adjacent to any x_i with $i \in \{2, 3, 4, 5\}$. Hence, as G has diameter 2, there exists a vertex $z \in N_1$ with $z \neq y$, such that z is adjacent to x_3 and v. We assume without loss of generality that $c(x_3) = 3$ and thus $c(x_2) = c(x_4) = 2$ and thus $c(x_5) = 3$. As $L'_c(v) = \{1, 2, 3\}$ and $c(x_3) = 3$, we find that $L'_c(z) = \{1, 2\}$. Hence, z is not adjacent to any vertex of $\{x_1, x_2, x_4\}$. Now the set $\{x_1, x_2, x_3, z, v, y\}$ forms a cycle on six vertices. As G is C_6-free, this cycle cannot be induced. Hence, the above implies that y and z must be adjacent; see also Figure 3.

As G has diameter 2, there exists a vertex $w \in N_1$ that is adjacent to x_4 and v. As both y and z are not adjacent to x_4, we find that $w \notin \{y, z\}$. As $L'_c(v) = \{1, 2, 3\}$ and $c(x_4) = 2$, we find that $L'_c(w) = \{1, 3\}$. As $c(x_1) = 1$ and $c(x_3) = c(x_5) = 3$, the latter implies that w is not adjacent to any vertex of $\{x_1, x_3, x_5\}$. Consequently, w must be adjacent to y, as otherwise the 6-vertex cycle with vertex set $\{x_1, x_5, x_4, w, v, y\}$ would be induced, contradicting the C_6-freeness of G. We refer again to Figure 3 for a display of the situation.

If w and z are adjacent, then $\{v, w, y, z\}$ induces a K_4, contradicting the K_4-freeness of G. Hence, w and z are not adjacent. Then $\{v, w, y, z\}$ induces a diamond, in which w and z are the two non-adjacent vertices. However, as $L'_c(w) = \{1, 3\}$ and $L'_c(z) = \{1, 2\}$, our algorithm would have applied Rule 4. This would have resulted in lists of w and z that are both equal to $\{1, 3\} \cap \{1, 2\} = \{1\}$. Hence, we obtained a contradiction and conclude that (G, L) is N_0-terminal. $\square$

Theorem 7 is proven in a similar way as Theorem 6 and we omit its proof.

Theorem 7. List 3-Colouring *can be solved in polynomial time for* (C_4, C_7)-*free graphs of diameter* 2.

4 The Extended Propagation Algorithm and Two Results

For our next two results, we need a more sophisticated method. Let (G, L) be an instance of List 3-Colouring. Let p be some positive constant. We consider each set $N_0 \subseteq V(G)$ of size at most p and perform a full N_0-propagation. Afterwards we say that we performed a *full p-propagation*. We say that (G, L) is *p-terminal* if after the full p-propagation one of the following cases hold:

1. for some $N_0 \subseteq V(G)$ with $|N_0| \leq c$, there is an L-promising N_0-precolouring c, such that the propagation algorithm returns `yes`; or
2. for every set $N_0 \subseteq V(G)$ with $|N_0| \leq c$ and every L-promising N_0-precolouring c, the propagation algorithm returns `no`.

We can now prove the following lemma.

Lemma 4. *Let (G, L) be an instance of* LIST 3-COLOURING *and $p \geq 1$ be some constant. Performing a full p-propagation takes polynomial time. Moreover, if (G, L) is p-terminal, then we have solved* LIST 3-COLOURING *on instance (G, L).*

Proof. For every set $N_0 \subseteq V(G)$, a full N_0-propagation takes polynomial time by Lemma 3. Then the first statement of the lemma follows from this observation and the fact that we need to perform $O(n^p)$ full N_0-propagations, which is a polynomial number, as p is a constant.

Now suppose that (G, L) is p-terminal. First assume that for some $N_0 \subseteq V(G)$ with $|N_0| \leq c$, there exists an L-promising N_0-precolouring c, such that the propagation algorithm returns `yes`. Then (G, L) is a yes-instance due to Lemma 2. Now assume that for every set $N_0 \subseteq V(G)$ with $|N_0| \leq c$ and every L-promising N_0-precolouring c, the propagation algorithm returns `no`. Then (G, L) is a no-instance. This follows from Lemma 2 combined with the observation that if (G, L) was a yes-instance, the restriction of a colouring c that respects L to any set N_0 of size at most p would be an L-promising N_0-precolouring of G. □

In our next two algorithms, we perform a full p-propagation for some appropriate constant p. If we find that an instance (G, L) is p-terminal, then we are done by Lemma 4. In the other case, we exploit the new information on the structure of G that we obtain from the fact that (G, L) is not p-terminal. We omit the proof of the first theorem.

Theorem 8. LIST 3-COLOURING *can be solved in polynomial time for (C_4, C_8)-free graphs of diameter 2.*

Theorem 9. LIST 3-COLOURING *can be solved in polynomial time for (C_4, C_9)-free graphs of diameter 2.*

Proof. Let $G = (V, E)$ be a (C_4, C_9)-free graph of diameter 2 with a list 3-assignment L. If G is C_7-free, then we apply Theorem 7. If G contains a K_4, then G is not 3-colourable and hence, (G, L) is a no-instance of LIST 3-COLOURING. We check these properties in polynomial time. So, from now on, we assume that G is a K_4-free graph that contains at least one induced cycle on seven vertices.

We set $p = 7$ and perform a full p-propagation. This takes polynomial time by Lemma 2. By the same lemma, we have solved LIST 3-COLOURING on (G, L) if (G, L) is p-terminal. Suppose we find that (G, L) is not p-terminal.

We first prove the following claim.

Claim 1. For each induced 7-vertex cycle C, the propagation algorithm returned `no` *for every L-promising $V(C)$-colouring c that assigns the same colour i on two vertices of C that have a common neighbour on C and that gives every other vertex of C a colour different from i.*

We prove Claim 1 as follows. Consider an induced 7-vertex cycle C, say with vertex set $N_0 = \{x_1, \dots, x_7\}$ in this order. Let N_1 be the set of vertices that do not belong to C but that are adjacent to at least one vertex of C. Let $N_2 = V \setminus (N_0 \cup N_1)$ be the set of remaining vertices. Let c be an L-promising $V(C)$-colouring that assigns two vertices of C with a common neighbour on C the same colour, say $c(x_1) = 1$ and $c(x_3) = 1$, and moreover, that assigns every vertex x_i with $i \in \{2, 4, 5, 6, 7\}$ colour $c(x_i) \neq 1$.

For contradiction, suppose that a full c-propagation does not yield a `no` output. As (G, L) is not p-terminal, this means that we obtained the c-propagated list assignment L'_c. By definition of L'_c we find that G contains a vertex v with $L'_c(v) = \{1, 2, 3\}$. Then $v \notin N_0$, as every $u \in N_0$ has $L'_c(u) = \{c(u)\}$. Moreover, $v \notin N_1$, as vertices in N_1 have a list of size at most 2 after applying Rule 3. Hence, we find that $v \in N_2$.

As G has diameter 2, there exist a vertex $y \in N_1$ that is adjacent to both v and x_1. Then y is not adjacent to any x_i with $i \in \{2, 4, 5, 6, 7\}$; in that case y would have a list of size 1 (as each x_i other than x_1 and x_3 is coloured 2 or 3) meaning that $L'_c(v)$ would have size at most 2. Hence, y is not adjacent to x_3 either, as otherwise $\{y, x_1, x_2, x_3\}$ would induce a C_4. As G has diameter 2, this means that there exists a vertex $y' \in N_1$ with $y' \neq y$ such that y' is adjacent to both v and x_3. By the same arguments we used for y', we find that x_3 is the only neighbour of y' on C.

If yy' is an edge then, by Rule 5, v would have had list $\{1\}$ instead of $\{1, 2, 3\}$. Hence, y and y' are not adjacent. However, now $\{y, v, y', x_3, x_4, x_5, x_6, x_7, x_1\}$ induces a C_9, a contradiction; see also Figure 4. This proves Claim 1.

Claim 1 tells us that if G has a colouring c respecting L, then c only gives the same colour to two vertices x and x' that are of distance 2 on some induced 7-vertex cycle C if there is a third vertex x'' that is of distance 2 from either x or x' on C with $c(x'') = c(x') = c(x)$. Hence, we can safely use the following new rule, whose execution takes polynomial time (in this rule, $c(x_1) = c(x_6)$ is not possible: view x_1 as x and x_6 as x' and note that x'' can neither be x_3 or x_4).

Rule-C7. ($\mathbf{C_7}$ colour propagation) Let C be an induced cycle on seven vertices $x_1, x_2, \dots, x_7$ in that order. If $|L(x_i)| = 1$ for $i \in \{1, 2, 3, 4\}$, $L(\{x_1, x_2, x_3\}) = \{1, 2, 3\}$, $L(x_4) = L(x_2)$, and $L(x_1) \subseteq L(x_6)$, then set $L(x_6) := \{1, 2, 3\} \setminus L(x_1)$ (so $L(x_6)$ gets size at most 2).

We now consider an induced 7-vertex cycle C in G, say on vertices $x_1, \dots, x_7$ in that order. Then either one colour appear once on C, or two colours appear exactly twice on C, with distance 3 from each other on C. Hence, we may assume

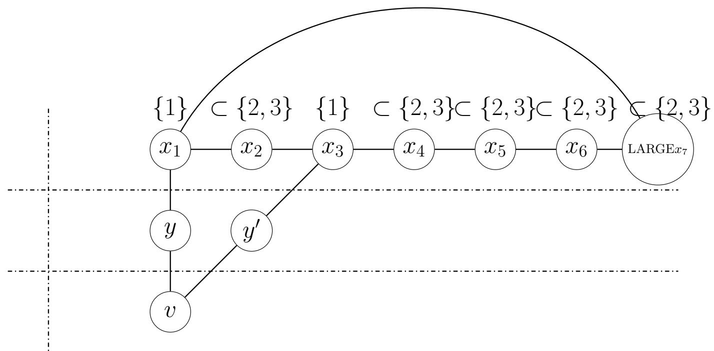

Fig. 4. The situation that is described in Claim 1 in the proof of Theorem 9. The set $\{x_1, y, v, y', x_3, x_4, x_5, x_6, x_7\}$ induces C_9, which is not possible.

without loss of generality that if G has a colouring c that respects L, then one of the following holds for such a colouring c (see also Figures 5 and 6):

(1) $c(x_1) = 1$, $c(x_2) = 2$, $c(x_3) = 3$, $c(x_4) = 2$, $c(x_5) = 3$, $c(x_6) = 2$, $c(x_7) = 3$; or
(2) $c(x_1) = 1$, $c(x_2) = 2$, $c(x_3) = 3$, $c(x_4) = 1$, $c(x_5) = 3$, $c(x_6) = 2$, $c(x_7) = 3$.

We let again $N_0 = \{x_1, \ldots, x_7\}$, N_1 be the set of vertices that do not belong to C but that are adjacent to at least one vertex of C, and $N_2 = V \setminus (N_0 \cup N_1)$ be the set of remaining vertices. We do a full c-propagation but now we also include the exhaustive use of Rule-C7. By combining Lemma 2 with the observation that Rule-C7 runs in polynomial time and reduces the list size of at least one vertex, this takes polynomial time. By combining the same lemma with the fact that Rule-C7 is safe (due to Claim 1) and the above observation that every L-respecting colouring of G coincides with c on N_0 (subject to colour permutation), we are done if we can prove that the propagation algorithm either outputs `yes` or `no`. We show that this is the case for each of the two possibilities (1) and (2) of c.

For contradiction, assume that the propagation algorithm returns `unknown`. Then we obtained the c-propagated list assignment L'_c. By definition of L'_c we find that G contains a vertex v with $L'_c(v) = \{1, 2, 3\}$. Then $v \notin N_0$, as every $u \in N_0$ has $L'_c(u) = \{c(u)\}$. Moreover, $v \notin N_1$, as vertices in N_1 have a list of size at most 2 after applying Rule 3. Hence, we find that $v \in N_2$. We now need to distinguish between the two possibilities of c.

Case 1 $c(x_1) = 1$, $c(x_2) = 2$, $c(x_3) = 3$, $c(x_4) = 2$, $c(x_5) = 3$, $c(x_6) = 2$, $c(x_7) = 3$
As G has diameter 2, there exists a vertex $y \in N_1$ that is adjacent to x_1 and v. Hence, y is not adjacent to any vertex in $\{x_2, \ldots, x_7\}$; otherwise y would have a list of size 1 due to Rule 3, and by the same rule, v would have a list of size 2. As G has diameter 2, there exists a vertex $y' \in N_1$ that is adjacent to x_4 and v. By the same arguments as above, y' is not adjacent to any vertex of $\{x_1, x_3, x_5, x_7\}$. The latter, together with the C_4-freeness of G, implies that y' is not adjacent to x_2 and x_6 either.

First suppose that $yy' \in E$. Then $\{x_1, x_7, x_6, x_5, x_4, y', y\}$ induces a C_7; see also Figure 5. As $c(x_1) = 1$, $c(x_7) = 3$, $c(x_6) = 2$ and $c(x_5) = 3$, we find that $L_c(\{x_1, x_7, x_6\}) = \{1, 2, 3\}$ and $L_c(x_5) = L_c(x_7)$. Then $1 \notin L_c(y')$, as otherwise the propagation algorithm would have applied Rule-C7. Moreover, $2 \notin L_c(y')$, as otherwise the propagation algorithm would have applied Rule 3. Hence, $L_c(y') = \{3\}$. However, then $|L_c(v)| \leq 2$, again due to Rule 3, a contradiction.

Now suppose that $yy' \notin E$. Then $\{x_1, x_2, x_3, x_4, y', v, y\}$ induces a C_7. As $c(x_1) = 1$, $c(x_2) = 2$, $c(x_3) = 3$, $c(x_4) = 2$, we find that $L_c(\{x_1, x_2, x_3\}) = \{1, 2, 3\}$ and $L_c(x_4) = L_c(x_2)$. Then $1 \notin L_c(v)$ due to Rule-C7. This is a contradiction, as we assumed $L_c(v) = \{1, 2, 3\}$. We conclude that the propagation algorithm returned either `yes` or `no`.

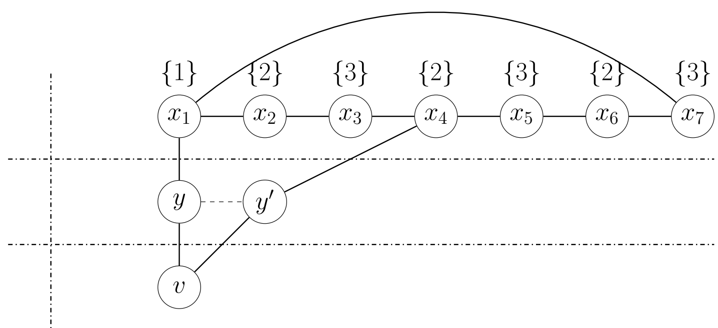

Fig. 5. The situation that is described in Case 1 in the proof of Theorem 9. If the edge yy' exists, then $\{x_1, x_7, x_6, x_5, x_4, y', y\}$ induces a C_7 to which Rule-C7 should have been applied. Otherwise the vertices $\{x_1, x_2, x_3, x_4, y', v, y\}$ induce such a C_7.

Case 2 $c(x_1) = 1$, $c(x_2) = 2$, $c(x_3) = 3$, $c(x_4) = 1$, $c(x_5) = 3$, $c(x_6) = 2$, $c(x_7) = 3$
As G has diameter 2, there is a vertex $y \in N_1$ adjacent to x_3 and v. Hence, y is not adjacent to any vertex in $\{x_1, x_2, x_4, x_6\}$; otherwise y would have a list of size 1 due to Rule 3, and by the same rule, v would have a list of size 2. As $yx_4 \notin E$, we find that $yx_5 \notin E$ either; otherwise $\{y, x_3, x_4, x_5\}$ induces a C_4. As G has diameter 2, this means there is a vertex $y' \in N_1 \setminus \{y\}$ adjacent to x_5 and v. By the same arguments as above, y' is not adjacent to any vertex of $\{x_1, x_2, x_4, x_6\}$. As G is C_4-free, the latter implies that $y'x_3 \notin E$ and $y'x_7 \notin E$.

If $yy' \in E$, then v would have a list of size at most 2 due to Rule 5. Hence $yy' \notin E$. If $yx_7 \notin E$, this means that $\{x_1, x_2, x_3, y, v, y', x_5, x_6, x_7\}$ induces a C_9, which is not possible. Hence, $yx_7 \in E$.

To summarize, we found that v has two distinct neighbours y and y', where y has exactly two neighbours on C, namely x_3 and x_7, and y' has exactly one neighbour on C, namely x_5. As G has diameter 2, this means that there exists a vertex $z \in N_1$ with $z \notin \{y, y'\}$ that is adjacent to x_6 and v. Then z is not adjacent to any vertex of $\{x_1, x_3, x_4, x_5, x_7\}$, as otherwise z would have a list

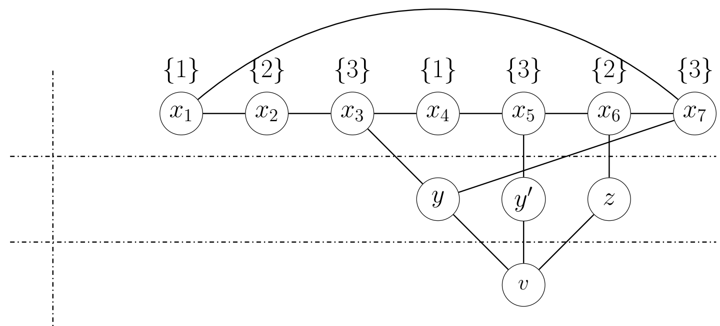

Fig. 6. The situation that is described in Case 2 in the proof of Theorem 9. The set $\{x_6, x_5, x_4, x_3, y, v, z\}$ induces a C_7 to which Rule-C7 should have been applied.

of size 1 due to Rule 3, and by the same rule, v would have a list of size 2. If $zy \in E$, then $\{y, z, x_6, x_7\}$ induces a C_4, which is not possible. Hence $zy \notin E$.

From the above, we find that $\{x_6, x_5, x_4, x_3, y, v, z\}$ induces a C_7; see also Figure 6. As $c(x_6) = 2$, $c(x_5) = 3$, $c(x_4) = 1$ and $c(x_3) = 3$, we find that $L_c(\{x_6, x_5, x_4\}) = \{1, 2, 3\}$ and $L_c(x_3) = L_c(x_5)$. Then $2 \notin L_c(v)$, due to Rule-C7. Hence, $|L_c(v)| \leq 2$, a contradiction. We conclude that the propagation algorithm returned either `yes` or `no` in Case 2 as well. □

5 Conclusions

We proved that 3-Colourability is polynomial-time solvable for several subclasses of diameter 2 that are characterized by forbidding one or two small induced cycles. In order to do this we used a unified framework of propagation rules, which allowed us to exploit the diameter-2 property of the input graph. Our current techniques need to be extended to obtain further results (in particular, we cannot currently handle the increasing number of different 3-colourings of induced cycles of length larger than 9).

As open problems we pose: determine the complexity of 3-Colouring and List 3-Colouring for graphs of diameter 2; C_t-free graphs of diameter 2 for $s \in \{3, 4, 7, 8, \ldots\}$; and (C_4, C_t)-free graphs of diameter 2 for $t \geq 10$. We also note that the complexity of k-Colouring for $k \geq 4$ and Colouring is still open for C_3-free graphs of diameter 2 (see also [19]).

Finally, we turn to the class of graphs of diameter 3. The construction of Mertzios and Spirakis [20] for proving that 3-Colouring is NP-complete for C_3-free graphs of diameter 3 appears to contain not only induced subdivided stars of arbitrary diameter and with an arbitrary number of leaves but also induced cycles of arbitrarily length $s \geq 4$. Hence, we pose as open problems: determine the complexity of 3-Colouring and List 3-Colouring for C_t-free graphs of diameter 3 for $t \geq 4$ and (C_4, C_t)-free graphs for $t \in \{3, 5, 6, \ldots\}$.

References

1. N. Alon. Restricted colorings of graphs. *Surveys in combinatorics, London Mathematical Society Lecture Note Series*, 187:1–33, 1993.
2. M. Bodirsky, J. Kára, and B. Martin. The complexity of surjective homomorphism problems - a survey. *Discrete Applied Mathematics*, 160(12):1680–1690, 2012.
3. F. Bonomo, M. Chudnovsky, P. Maceli, O. Schaudt, M. Stein, and M. Zhong. Three-coloring and list three-coloring of graphs without induced paths on seven vertices. *Combinatorica*, 38(4):779–801, 2018.
4. H. Broersma, F. V. Fomin, P. A. Golovach, and D. Paulusma. Three complexity results on coloring P_k-free graphs. *European Journal of Combinatorics*, 34(3):609–619, 2013.
5. M. Chudnovsky. Coloring graphs with forbidden induced subgraphs. *Proc. ICM 2014*, IV:291–302, 2014.
6. R. M. Damerell. On Moore graphs. *Proc. Cambridge Philosophical Society*, 74:227–236, 1973.
7. K. Edwards. The complexity of colouring problems on dense graphs. *Theoretical Computer Science*, 43:337–343, 1986.
8. T. Emden-Weinert, S. Hougardy, and B. Kreuter. Uniquely colourable graphs and the hardness of colouring graphs of large girth. *Combinatorics, Probability and Computing*, 7(04):375–386, 1998.
9. P. A. Golovach, M. Johnson, D. Paulusma, and J. Song. A survey on the computational complexity of colouring graphs with forbidden subgraphs. *Journal of Graph Theory*, 84(4):331–363, 2017.
10. P. A. Golovach, D. Paulusma, and J. Song. Coloring graphs without short cycles and long induced paths. *Discrete Applied Mathematics*, 167:107–120, 2014.
11. A. J. Hoffman and R. R. Singleton. On Moore graphs with diameter 2 and 3. *IBM Journal of Research and Development*, 5:497–504, 1960.
12. I. Holyer. The NP-completeness of edge-coloring. *SIAM Journal on Computing*, 10(4):718–720, 1981.
13. T. R. Jensen and B. Toft. *Graph coloring problems*. John Wiley & Sons, 1995.
14. T. Klimošová, J. Malík, T. Masařík, J. Novotná, D. Paulusma, and V. Slívová. Colouring $(P_r + P_s)$-free graphs. *Proc. ISAAC 2018, LIPIcs*, 123:5:1–5:13, 2018.
15. J. Kratochvíl, Zs. Tuza, and M. Voigt. New trends in the theory of graph colorings: choosability and list coloring. *Proc. DIMATIA-DIMACS Conference*, 49:183–197, 1999.
16. L. Lovász. Coverings and coloring of hypergraphs. *Proc. 4th Southeastern Conference on Combinatorics, Graph Theory, and Computing, Utilitas Math.* pages 3–12, 1973.
17. V. V. Lozin and M. Kaminski. Coloring edges and vertices of graphs without short or long cycles. *Contributions to Discrete Mathematics*, 2(1), 2007.
18. V. V. Lozin and D. S. Malyshev. Vertex coloring of graphs with few obstructions. *Discrete Applied Mathematics*, 216:273–280, 2017.
19. B. Martin, D. Paulusma, and S. Smith. Colouring H-free graphs of bounded diameter. *Proc. MFCS 2019, LIPIcs*, 138:14:1–14:14, 2019.
20. G. B. Mertzios and P. G. Spirakis. Algorithms and almost tight results for 3-colorability of small diameter graphs. *Algorithmica*, 74(1):385–414, 2016.
21. D. Paulusma. Open problems on graph coloring for special graph classes. *Proc. WG 2015, LNCS*, 9224:16–30, 2015.
22. M. Pilipczuk, M. Pilipczuk, and P. Rzążewski. Quasi-polynomial-time algorithm for independent set in P_t-free and $C_{\geq t}$-free graphs via shrinking the space of connecting subgraphs. *CoRR*, abs/2009.13494, 2020.

23. B. Randerath and I. Schiermeyer. Vertex colouring and forbidden subgraphs – a survey. *Graphs and Combinatorics*, 20(1):1–40, 2004.
24. A. Rojas and M. Stein. 3-colouring P_t-free graphs without short odd cycles. *CoRR*, abs/2008.04845, 2020.
25. T. J. Schaefer. The complexity of satisfiability problems. *Proc. STOC 1978*, pages 216–226, 1978.
26. Z. Tuza. Graph colorings with local constraints - a survey. *Discussiones Mathematicae Graph Theory*, 17(2):161–228, 1997.