Towards a Complexity Dichotomy for Colourful Components Problems on k-caterpillars and Small-Degree Planar Graphs

Janka Chlebíková and Clément Dallard[✉]

School of Computing, University of Portsmouth, Portsmouth, UK
{janka.chlebikova,clement.dallard}@port.ac.uk

Abstract. A connected component of a vertex-coloured graph is said to be *colourful* if all its vertices have different colours, and a graph is colourful if all its connected components are colourful. Given a vertex-coloured graph, the COLOURFUL COMPONENTS problem asks whether there exist at most p edges whose removal makes the graph colourful, and the COLOURFUL PARTITION problem asks whether there exists a partition of the vertex set with at most p parts such that each part induces a colourful component. We study the problems on k-caterpillars (caterpillars with hairs of length at most k) and explore the boundary between polynomial and NP-complete cases. It is known that the problems are NP-complete on 2-caterpillars with unbounded maximum degree. We prove that both problems remain NP-complete on binary 4-caterpillars and on ternary 3-caterpillars. This answers an open question regarding the complexity of the problems on trees with maximum degree at most 5. On the positive side, we give a linear time algorithm for 1-caterpillars with unbounded degree, even if the backbone is a cycle, which outperforms the previous best complexity for paths and widens the class of graphs. Finally, we answer an open question regarding the complexity of COLOURFUL COMPONENTS on graphs with maximum degree at most 5. We show that the problem is NP-complete on 5-coloured planar graphs with maximum degree 4, and on 12-coloured planar graphs with maximum degree 3. Since the problem can be solved in polynomial-time on graphs with maximum degree 2, the results are the best possible with regard to the maximum degree.

Keywords: Colorful component · Caterpillar · Binary tree · Planar subcubic graph

1 Introduction

A *coloured graph* is a graph whose vertices are (not necessarily properly) coloured. A connected component of a coloured graph is a *colourful component*

if all its vertices have different colours. A graph is said to be colourful if all its connected components are colourful.

In this paper we focus on two closely related problems where a coloured graph and a positive integer p are given as inputs: the COLOURFUL COMPONENTS problem asks if there exist at most p edges whose removal makes the graph colourful; the COLOURFUL PARTITION problem is to decide if there exists a partition of the vertex set with at most p parts such that each part induces a colourful component in the graph.

One key problem in comparative genomics is to partition a set of genes into orthologous genes, which are sets of genes in different species that have evolved through speciation events only, *i.e.* originated by vertical descent from a single gene in the last common ancestor. The problem has been modelled as a graph problem where orthologous genes translate into colourful components in the graph [1,15]. The vertices of the graph represent the genes, and a colour is given to each vertex to symbolise the specie the corresponding gene belongs to. An edge between two vertices is present in the graph if the two corresponding genes are (sufficiently) similar. The quality of a partition of a set of genes into orthologous genes can be expressed in different ways. Minimising the number of similar genes in different subsets of the partition is a well studied variant [4,5,8,13,15], and it corresponds to minimising the number of edges between the colourful components (as in COLOURFUL COMPONENTS). Alternatively, one can ask for a partition of minimum size, *i.e* which contains the minimum number of orthologous genes, or equivalently the minimum number of colourful components [1,5,6] (as in COLOURFUL PARTITION). Another variant, not studied in this paper, considers the objective function that maximises the number of edges in the transitive closure [1,6,13].

Now, we give the formal definitions of the problems considered herein.

COLOURFUL COMPONENTS

Input: A vertex-coloured graph $G = (V, E)$, a positive integer p.
Question: Are there at most p edges in E whose removal makes G colourful?

COLOURFUL PARTITION

Input: A vertex-coloured graph $G = (V, E)$, a positive integer p.
Question: Is there a partition of V with at most p parts s.t. each part induces a colourful component in G?

It is interesting to notice the similarities between COLOURFUL COMPONENTS and the MULTICUT [3,12] and MULTI-MULTIWAY CUT [2] problems. In the MULTICUT problem, a graph and a set of pairs of vertices are given and the goal is to minimise the number of edges to remove in order to disconnect each pair of vertices. In the MULTI-MULTIWAY CUT problem, a graph and sets of vertices are given and the goal is to minimise the number of edges to remove in order to disconnect all paths between vertices from the same vertex set. Thus, COLOURFUL COMPONENTS is a special case where the sets of vertices form a partition.

Both COLOURFUL COMPONENTS and COLOURFUL PARTITION problems can be compared to the GRAPH MOTIF problem [7]. This problem takes a coloured

graph and a multiset of colours M (the motif) as input, and the goal is to determine whether there exists a connected subgraph S such that the multiset of colours used by the vertices in S corresponds exactly to M. If M is a set (where each colour appears at most once), then M is said to be *colourful*.

In this paper, all graphs are simple. We assume that a coloured graph $G = (V, E)$ is always associated with a colouring function c from V to a set of colours, hence for each vertex $u \in V$, $c(u)$ is the colour of the vertex u. The *colour multiplicity* of G corresponds to the maximum number of occurrence of any colour in the graph. If G contains at most ℓ colours we say that G is an ℓ-coloured graph. To simplify the notations, we may say that a vertex u belongs to a path P in G if there exists an edge $e \in P$ with $u \in e$. A path P in G between two vertices u and v is called a *bad path* if $c(u) = c(v) = \gamma$ and u, v are the only two vertices of colour γ in P. Hence, a connected component is colourful if and only if it does not contain a bad path. Lastly, given a set of edges $S \subseteq E$, we denote by $G - S$ the graph $(V, E \setminus S)$, and for a vertex $u \in V$, $N[u]$ is the closed neighbourhood of u.

A *k-caterpillar*, also commonly called a *caterpillar with hairs of length at most k* [10], is a tree in which all the vertices are within distance k of a central path, called the *backbone*. Similarly, we define a *cyclic k-caterpillar* as a k-caterpillar whose backbone is a chordless cycle. Note that 2-caterpillars are also known as *lobster graphs*.

Observe that, on a tree, there is a solution to COLOURFUL COMPONENTS with p edges if and only if there is a solution to COLOURFUL PARTITION with $p + 1$ parts. However, this is not the case on general graphs [5]. Both problems are known to be NP-complete on subdivided stars [6], trees of diameter at most 4 [4], and trees with maximum degree 6 [5]. Trees of diameter at most 4 are in fact a subclass of 2-caterpillars, so both problems are NP-complete on 2-caterpillars when the maximum degree is unbounded. In Sect. 2.1, we prove that COLOURFUL COMPONENTS and COLOURFUL PARTITION are NP-complete on binary 4-caterpillars and on ternary 3-caterpillar, hence with the maximum degree at most 3 or 4. This answers an open question, proposed in [5], regarding the complexity of the problems on trees with maximum degree at most 5. Nonetheless, we propose a linear time algorithm for COLOURFUL COMPONENTS and COLOURFUL PARTITION on 1-caterpillars and cyclic 1-caterpillars with unbounded degree in Sect. 2.2. This result improves the complexity of the known quadratic-time algorithm for paths [6] and widens the class of graphs. We, therefore, obtain a complete complexity dichotomy for the problems on k-caterpillars with regard to k and the maximum degree in the graph.

We also consider the complexity of COLOURFUL COMPONENTS in planar graphs with small degree. It is known that the problem is NP-complete on 3-coloured graphs with maximum degree 6 [4], while COLOURFUL PARTITION is NP-complete on 3-coloured 2-connected planar graphs with maximum degree 3 [5]. However, it was an open question whether COLOURFUL COMPONENTS is NP-complete on ℓ-coloured graphs with maximum degree at most 5. In Sect. 3, we answer that question and show that COLOURFUL COMPONENTS is NP-complete

on 5-coloured planar graphs with maximum degree 4 and on 12-coloured planar graphs with maximum degree 3. As COLOURFUL COMPONENTS is polynomial-time solvable on graphs with maximum degree 2, our result is the best possible with regard to the maximum degree.

2 Complexity on k-caterpillars

In this section, we focus on the complexity of COLOURFUL COMPONENTS and COLOURFUL PARTITION on k-caterpillars, depending on the value of k and the maximum degree of the graphs.

2.1 NP-completeness

First, we show that COLOURFUL COMPONENTS and COLOURFUL PARTITION are NP-complete on binary 4-caterpillars and ternary 3-caterpillars. We recall that a binary tree (resp. ternary) is a rooted tree in which each vertex has no more than two children (resp. three children). We propose a reduction from 3-SAT with at most four occurrence of each variable, known as 3, 4-SAT, which is proved NP-complete in [14].

Construction 1. Consider an instance ϕ of 3, 4-SAT, that is, a set of m clauses $C_1, C_2, \ldots, C_m$ on n variables $x_1, x_2, \ldots, x_n$, where each clause contains exactly three literals and where each variable appears at most four times.

For each variable x_i, we define a *variable gadget*. Firstly, we create two vertices labelled x_i and $\bar{x}_i$, which are the roots of two binary trees T_{x_i} and $T_{\bar{x}_i}$, respectively. If a clause C_j contains the literal x_i, then we create a vertex labelled $x_{i,j}$ in T_{x_i}. Similarly, if a clause C_j contains the literal $\bar{x}_i$, then there we create a vertex labelled $\bar{x}_{i,j}$ in $T_{\bar{x}_i}$. All created vertices are connected in such a way that T_{x_i} and $T_{\bar{x}_i}$ are binary trees of depth at most 2. We assume that all the vertices in the trees, except for x_i and $\bar{x}_i$, correspond to one literal in a clause. Finally, we connect x_i and $\bar{x}_i$ to a same new vertex r_{x_i}. Notice that the gadget is a binary tree of depth at most 3 (see Fig. 1).

For each clause C_j, we define a *clause gadget*. Let ℓ_1, ℓ_2 and ℓ_3 be the literals in C_j. We create four vertices y_j, y'_j, z_j and z'_j, three vertices labelled $\ell_{1,j}$, $\ell_{2,j}$ and $\ell_{3,j}$ representing the literals in C_j, and one extra vertex r_{C_j}. Then, we add the edges $\{\ell_{1,j}, y_j\}$, $\{\ell_{2,j}, y'_j\}$, $\{\ell_{3,j}, z'_j\}$, $\{y_j, z_j\}$, $\{y'_j, z_j\}$, and the edges $\{z_j, r_{C_j}\}$ and $\{z'_j, r_{C_j}\}$. Notice that the gadget is a binary tree of depth 3 (see Fig. 1).

Now, we describe two slightly different ways to obtain a tree T containing all the variable and clause gadgets.

– To get T as a binary 4-caterpillar, create a central path with $n + m$ new vertices and connect each of the $n + m$ vertices r_{x_i} and r_{C_j} to a different vertex of the central path.
– To get T as a ternary 3-caterpillar, connect the $n + m$ vertices r_{x_i} and r_{C_j} together (in any order) to create a central path.

In both cases, the central path corresponds to the backbone of T. We set the root r of T such that it belongs to the backbone of T and has minimum degree, hence two or three children, if T is a binary or ternary caterpillar, respectively.

Finally, we assign a colour to each vertex in T. For each gadget representing a variable x_i, let $c(x_i) = c(\bar{x}_i)$ be a new colour. Also, for each vertex $\widetilde{x}_{i,j} \in \{x_{i,j}, \bar{x}_{i,j}\}$, let $c(\widetilde{x}_{i,j})$ be a new colour. Then, for each gadget representing a clause C_j, set the colour of each leaf $\ell_{k,j}$ such that if $\ell_{k,j} = x_i$, then $c(\ell_{k,j}) := c(x_{i,j})$, but if $\ell_{k,j} = \bar{x}_i$, then $c(\ell_{k,j}) := c(\bar{x}_{i,j})$. Furthermore, let $c(y_j) = c(y'_j)$ and $c(z_j) = c(z'_j)$ be two new colours. Lastly, all the vertices in T which do not belong to any gadget are given new colours. Notice that there are no such vertices if T is a 3-caterpillar. Obviously, in both cases, T is a coloured caterpillar with colour-multiplicity 2.

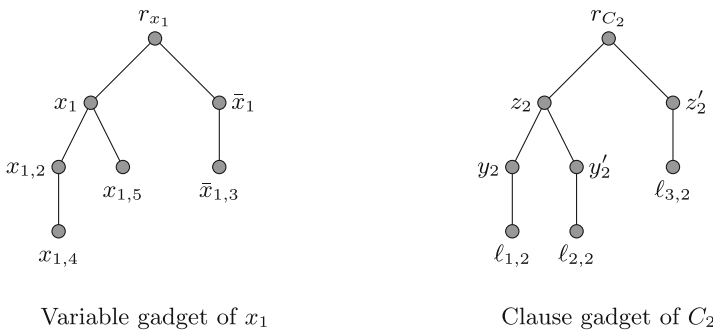Note that Construction 1 can be done in polynomial time.

Variable gadget of x_1 Clause gadget of C_2

Fig. 1. Examples of gadgets used in Construction 1. On the left, the variable gadget of x_1, appearing as a positive literal in C_2, C_4 and C_5, and as a negative literal in C_3. On the right, the clause gadget of C_2.

Theorem 1. COLOURFUL COMPONENTS *and* COLOURFUL PARTITION *are* NP-*complete on coloured ternary 3-caterpillars with colour-multiplicity 2 and on coloured binary 4-caterpillars with colour-multiplicity 2.*

Proof. Obviously, the problem is in NP. Let ϕ be an instance of 3, 4-SAT with m clauses and n variables. We transform ϕ into a coloured tree T as described in Construction 1 such that T has colour multiplicity 2 and is a coloured binary 4-caterpillar or a coloured ternary 3-caterpillar. We claim that there is a solution to 3, 4-SAT on ϕ if an only if there is a set of exactly $n + 2m$ edges in T whose removal makes T colourful.

Let β be a solution to 3, 4-SAT on ϕ. We define the set of edges S as follows:

– For each variable x_i, the set S contains the edge $\{r_{x_i}, x_i\}$ if $x_i = True$ in β, or $\{r_{x_i}, \bar{x}_i\}$ if $x_i = False$ in β.

– For each clause C_j, the set S contains two edges: one from the path between y_j and y'_j, and one from the path between z_j and z'_j. Moreover, in $G - S$, the leaf $\ell_{k,j}$ which belongs to the same connected component as the vertex in r_{C_j} must correspond to (one of) the literal(s) satisfying the clause C_j in β.

Clearly, the set S contains $n + 2m$ edges. We denote by $\mathcal{F}$ the forest $T - S$, and by T' the connected component in $\mathcal{F}$ containing the root r. Obviously, two vertices of the same colour from a same variable gadget do not both belong to a same connected component of $\mathcal{F}$, and the same is true for a clause gadget. Also, note that two vertices of different variable gadgets do not have the same colour, and similarly for vertices of different clause gadgets. Lastly, observe that two vertices of two different gadgets belong to the same connected component if and only if they are connected through the backbone, which is in T'. Thus, if there exist two vertices of the same colour in a same connected component of $\mathcal{F}$, one must be from a variable gadget and the other one from a clause gadget, and they both necessarily belong to T'. Without loss of generality, consider $x_{i,j}$ from the variable gadget of x_i and $\ell_{k,j}$ from the clause gadget of C_j, such that $x_{i,j}, \ell_{k,j} \in T'$. To prove a contradiction, assume that $c(x_{i,j}) = c(\ell_{k,j})$. Note that the literal represented by $\ell_{k,j}$ is $x_{i,j}$, otherwise the two vertices would not have the same colour. Since $\ell_{k,j}$ is in T', it is connected to the vertex r_{C_j} of the clause gadget, hence $\ell_{k,j}$ satisfies the clause C_j. Therefore, x_i satisfies the clause C_j, and the variable $x_i = True$ in β. By construction, this implies that the edge $\{r_{x_i}, x_i\}$ belongs to S, and therefore that the subtree T_{x_i}, containing $x_{i,j}$, is not part of T', which is a contradiction.

Let S be a solution to COLOURFUL COMPONENTS on T such that $|S| = n + 2m$. Observe that one needs to remove at least one edge per variable gadget to put the vertices x_i and $\bar{x}_i$ into different connected components, and at least two edges per clause gadget, the first one to disconnect y_j and y'_j and the second one to disconnect z_j and z'_j. Since $|S| = n + 2m$, S must only contain n edges from variable gadgets and $2m$ edges from clause gadgets. We denote by T' the connected component of $T - S$ containing the root r. Notice that, for each variable gadget, either x_i or $\bar{x}_i$ belongs to T', but not both. Also, for each clause gadget, exactly one leaf $\ell_{k,j}$ belongs to T'. We construct the solution β to 3, 4-SAT on ϕ such that, for each variable gadget, if $\{r_{x_i}, x_i\} \in S$, then we set $x_i := True$ in β, and if $\{r_{x_i}, \bar{x}_i\} \in S$, then we set $x_i := False$ in β. To prove a contradiction, assume that there is a clause C_j which is not satisfied in ϕ with regard to β. Consider the leaf $\ell_{k,j} \in T'$ from the gadget clause of C_j, and assume without loss of generality that $\ell_{k,j} = x_i$. If C_j is not satisfied, then the variable $x_i := False$ in β. This means that S contains the edge $\{r_{x_i}, \bar{x}_i\}$, but not the edge $\{r_{x_i}, x_i\}$, and thus $x_{i,j} \in T'$. However, since $c(x_{i,j}) = c(\ell_{k,j})$, then S is not a solution for T, a contradiction. □

2.2   Polynomiality

Now, we show that COLOURFUL COMPONENTS and COLOURFUL PARTITION can be solved in linear time on 1-caterpillars with unbounded maximum degree, even

if the backbone is a chordless cycle. To simplify the notations, we use the term *general caterpillars* to denote both 1-caterpillars and cyclic 1-caterpillars.

We consider the vertices in the backbone as internal vertices of stars, hence vertices of degree 1 are the leaves of a star whose internal vertex belongs to the backbone. We assume that the edges and the vertices in the backbone are either linearly of cyclically ordered, if the backbone is a path or a cycle, respectively.

Remark 1. Consider a coloured general caterpillar. If two vertices of a star have the same colour, then at least one of these vertices is a leaf and it must belong to a different colourful component than the internal vertex of the star. Hence, a general caterpillar can be preprocessed in such a way that, for each such leaf, we add its adjacent edge to a set S_p. This procedure is repeated until there is no such leaf in $G - S_p$. At the end of the preprocessing, each star in $G - S_p$ is a *colourful star*, that is, only contains vertices with different colours.

If a coloured general caterpillar is not colourful, then it contains either one or at least two colours that appear more than once. We deal with these two cases independently in the following lemmas.

Lemma 1. COLOURFUL COMPONENTS *and* COLOURFUL PARTITION *can be solved in linear time in coloured general caterpillars where exactly one colour appears at least twice.*

Lemma 2. *Let G be a coloured general caterpillar with only colourful stars such that at least two colours appear at least twice in G. Then there exists an optimal solution S of* COLOURFUL COMPONENTS *in G such that $S \subseteq B$, where B is the backbone of G.*

Let G be a coloured general caterpillar with backbone B and only colourful stars. We say that a bad path P between two vertices of colour γ in G is a *colour-critical bad* path if and only if there is no other bad path P' between two vertices of colour γ such that $P' \cap B \subseteq P$. Hence, two colour-critical bad paths with endpoints of colour γ do not have any common edge in the backbone B.

Remark 2. Let G be a coloured general caterpillar with only colourful stars such that at least two colours appear twice. We denote by B the backbone of G. Lemma 2 guarantees that there exists an optimal solution S to COLOURFUL COMPONENTS on G such that $S \subseteq B$. It is clear that if each colour-critical bad path contains an edge in S and $S \subseteq B$, then S is a solution to COLOURFUL COMPONENTS on G. Hence, there is an optimal solution to COLOURFUL COMPONENTS that contains only edges in B that also belong to some colour-critical bad path.

Now, the idea is to define a circular-arc graph H (an intersection graph of a collection of arcs on the circle) based on the colour-critical bad paths of G. A minimum clique cover $\mathcal{Q}$ of H, which is a partition of the vertex set into a minimum number of cliques, can be obtained in linear time [9]. We show that $\mathcal{Q}$ can be translated back into an optimal solution to COLOURFUL COMPONENTS and COLOURFUL PARTITION on G in linear time.

Algorithm 1. From coloured general caterpillar to ordered pairs.

Input: $G = (V, E)$, an ℓ-coloured general caterpillar with only colourful stars
and backbone B.
Output: A, a multiset of ordered pairs of vertices.
`// Initialisation`
1 let $U := \{u \in e \mid e \in B\}$ be an ordered set of vertices, w.r.t. the order on B;
2 let A be an empty multiset of ordered pairs of vertices;
3 let L be an array of length ℓ initialised at $NULL$;
4 let $proceed := True$;
5 let $u \in U$ such that, if U is linearly ordered, then u is minimum in U;
`// if U is cyclically ordered, any u ∈ U can be taken`
6 let $end := NULL$;
`// Main procedure`
7 **while** $proceed$ **do**   `// O(n)`
8   **foreach** $v \in \{w \in N[u] \mid d(w) = 1 \text{ or } w = u\}$ **do**
9   **if** $L[c(v)] = NULL$ **then**
10           $end := u$;
11   **else if** $L[c(v)] \neq v$ **then**
12   add $(L[c(v)], u)$ to A;
13   **if** $u = end$ **then**
14             $proceed := False$;   `// U is cyclically ordered`
15         $L[c(v)] := v$;
16   **if** U is linearly ordered and u is maximum in U **then**
17         $proceed := False$;
18     $u := v$, such that v is the element after u in U, w.r.t. its order;
19 **return** A;

Lemma 3. *Let G be a coloured general caterpillar with only colourful stars,
and A be the multiset of pairs returned by Algorithm 1. Then there is a bijection
between the set of colour-critical bad paths in G and the multiset A.*

Proof. Let B be the backbone of $G = (V, E)$. A colour-critical bad path P from
a to b is detected in Algorithm 1 at Line 11, when b is found to have the same
colour γ as a (the last recorded vertex of colour γ). Let x be the internal vertex of
the star to which a belongs, and y for b, respectively. When b is considered in the
algorithm, the pair $(L[c(b)], y)$ is added to A at Line 12, and since $L[c(b)] = x$
then $(x, y) \in A$. Thus, the arc with endpoints $(x, y) \in A$ corresponds to the
colour-critical bad path P from a to b in G.

An ordered pair (x, y) in A refers to two vertices x and y in V that are
internal vertices of two stars. If such a pair exists, then there are two vertices a
and b with the same colour γ, such that a belongs to the same star as x and b
to the same star as y, and in the path P from a to b, with regard to the order
on B, there is no other vertex w with colour γ in a star whose internal vertex is
in P (since the last seen vertex of colour γ, before b, is $L[c(b)] = a$). Thus, the
path P is a colour-critical bad path and it corresponds to the pair (x, y) in A.□

Lemma 4. *Algorithm 1 runs in linear time.*

Proof. Let G be a coloured general caterpillar with only colourful stars and backbone B, and let A be the multiset of ordered pairs obtained by Algorithm 1 with input G.

In Algorithm 1, when a colour is detected for the first time at Line 9, the internal vertex u of the star is stored in the variable *end*. If the backbone is a chordless cycle, *i.e.* if G is a cyclic caterpillar, the second time that the vertex *end* is considered in the main loop the algorithm sets the variable *proceed* to false at Line 14. If the backbone is a path, *i.e.* if G is a caterpillar, the algorithm considers each vertex exactly once and sets the variable *proceed* to false at Line 17. Thus, Algorithm 1 runs in linear time.   □

Theorem 2. COLOURFUL COMPONENTS *and* COLOURFUL PARTITION *can be solved in linear time on coloured general caterpillars.*

Proof. Let $G = (V, E)$ be a coloured general caterpillar with backbone B. First, we prove that a solution to COLOURFUL COMPONENTS on G can be found in linear time. We apply the preprocessing to G, as defined in Remark 1, and denote by S_p the set of edges that have been removed. Hence, $G - S_p$ contains only colourful stars. If $G - S_p$ is colourful, then S_p is an optimal solution to COLOURFUL COMPONENTS. Otherwise, denote by $G' = (V', E')$ the connected component of $G - S_p$ which contains the backbone B. If G' contains exactly one colour that appears more than once, then according to Lemma 1 COLOURFUL COMPONENTS and COLOURFUL PARTITION can be solved in linear time. Therefore, we assume that G' contains at least two colours that appear at least twice. Let A be the multiset of ordered pairs obtained by Algorithm 1 with input G'.

According to Lemma 4, A can be obtained in linear time. According to Lemma 3, each ordered pair (x, y) in A corresponds to a colour-critical bad path P from x to y in G. These paths can be seen as arcs on the circle, which represent a circular-arc graph $H = (X, F)$. Let $\mathcal{Q}$ be a minimum clique cover of H obtained in linear time [9], and S' be an empty set of edges. Choose a clique $Q_i \in \mathcal{Q}$. From our construction of H, each vertex $u \in Q_i$ corresponds to a colour-critical bad path P_u in G. Let $D_i := \bigcap_{u \in Q_i} P_u$, and notice that $|D_i \cap B| > 0$. Then, choose an edge $e \in D_i \cap B$, and add e to S'. We claim that, once each clique in $\mathcal{Q}$ has been processed, thus once $|S'| = |\mathcal{Q}|$, the set S' is an optimal solution to COLOURFUL COMPONENTS on G. Notice that S' can be computed in linear time.

As stated before, each colour-critical bad path in G' maps to an ordered pair in A, which corresponds to a vertex of H. Hence, a clique Q_i in H corresponds to a set of colour-critical bad paths sharing a common subpath D_i. The set S' contains an edge in $D_i \cap B$ for each $Q_i \in \mathcal{Q}$, hence there is no colour-critical bad path in $G' - S'$. Since $S' \subset B$ and each coloured-minimal bad paths have an edge in S', then following Remark 2 S' is a solution to COLOURFUL COMPONENTS on G'. Moreover, since Q is optimal, S' is an optimal solution on G'. Let $S := S_p \cup S'$, and note that S is an optimal solution to COLOURFUL COMPONENTS on G.

Since $|E| \in \mathcal{O}(|V|)$, we can detect each connected component of $G - S$ in linear time (for instance, with a breadth-first search). Thus, we can construct the partition π of V such that each part corresponds to a connected component of $G - S$ in linear time. Obviously, π is a solution to COLOURFUL PARTITION on G. Since S is optimal, due to the structure of G, the partition π is optimal.   □

3   COLOURFUL COMPONENTS on Small-Degree Planar Graphs

In [4], the authors prove that COLOURFUL COMPONENTS is NP-complete even when restricted to 3-coloured graphs with maximum degree 6. Using a similar reduction from PLANAR 3-SAT, we show that the vertices of degree 6 can be replaced with gadgets only containing vertices of degree 4, or 3, if we relax the number of colours from 3 to 5, or 12, respectively.

An instance of PLANAR 3-SAT is a 3-CNF formula in which the bipartite graph of variables and clauses is planar. PLANAR 3-SAT has been proved NP-complete in [11].

Construction 2. Given an instance of PLANAR 3-SAT ϕ, that is a set of m clauses $C_1, C_2, \ldots, C_m$ on n variables, we construct the graph $G = (V, E)$ such that:

– For each variable x in ϕ, let m_x denotes the number of clauses in which x appears. We construct a cycle of length $4m_x$ in G with vertices $V_x := \{x_j^1, x_j^2, x_j^3, x_j^4 \mid x \in C_j\}$ with an arbitrary fixed cyclic ordering of the clauses containing x. The vertices are coloured alternatively with two colours c_o and c_e, that is, $c(x_j^1) = c(x_j^3) = c_o$ and $c(x_j^2) = c(x_j^4) = c_e$, for all j such that $x \in C_j$.
– For each clause C_j containing three variables p, q and r, we construct a clause gadget. We propose two types of gadgets:
  • The gadget $\mathcal{A}_j^4$ is made of a cycle of length 3, with vertices a_j^1, a_j^2 and a_j^3 such that each a_j^i is given colour i, different from c_o and c_e. We define how the vertices from V_p are connected to $\mathcal{A}_j^4$. If the variable p appears as a positive literal in C_j, connect the vertices p_j^1 to a_j^1 and p_j^2 to a_j^2. Otherwise, if p occurs as a negative literal, connect the vertices p_j^2 to a_j^1 and p_j^3 to a_j^2. Do the same for the variables q and r by connecting the corresponding vertices in V_q to a_j^2 and a_j^3, and the corresponding vertices in V_r to a_j^3 and a_j^1. Notice that the vertices in $\mathcal{A}_j^4$ have degree 4.
  • The gadget $\mathcal{A}_j^3$ is made of a cycle of length 9 with vertices labelled $a_j^1, \ldots, a_j^9$ and an additional vertex a_j^{10} connected to a_j^2, a_j^5 and a_j^8. We set the colour i to each vertex a_j^i, different from c_o and c_e. We define how the vertices from V_p are connected to $\mathcal{A}_j^3$. If the variable p appears as a positive literal in C_j, connect the vertices p_j^1 to a_j^1 and p_j^2 to a_j^3. Otherwise, the variable p occurs as a negative literal, connect the vertices p_j^2 to a_j^1 and p_j^3 to a_j^3. Do the same for the variables q and r by connecting the corresponding vertices in V_q to a_j^4 and a_j^6, and the corresponding vertices in V_r a_j^7 and a_j^9. Notice that the vertices in $\mathcal{A}_j^3$ have degree 3.
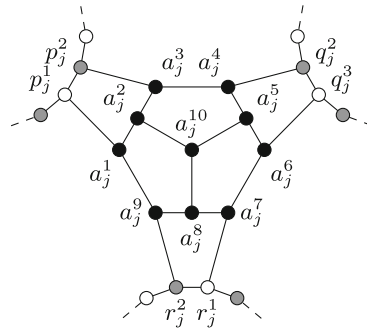
See Fig. 2 for an example of the gadgets.

Since the bipartite graph of variables and clauses of ϕ is planar and each vertex can be replaced by a clause or vertex gadget, with a correct cyclic ordering of the clauses for each variable, the resulting graph G is planar.

Note that Construction 2 can be done in polynomial time.
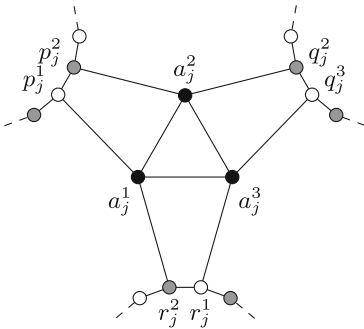
Gadget with vertices of degree 4.   Gadget with vertices of degree 3.

Fig. 2. Two possible clause gadgets of a clause $C_j := (p \vee \bar{q} \vee r)$. White vertices have colour c_o, grey vertices have colour c_e, and each a_j^i is given colour i, different from c_o and c_e.

Theorem 3. COLOURFUL COMPONENTS *is NP-complete on* 5-*coloured planar graphs with maximum degree* 4 *and on* 12-*coloured planar graphs with maximum degree* 3.

4   Conclusion

This paper proposes of complete dichotomy of the computational complexity of COLOURFUL COMPONENTS and COLOURFUL PARTITION on k-caterpillars. The NP-completeness of the problems on 2-caterpillars with unbounded degree demonstrates the inherent complexity of the problems. We prove that both problems remain NP-complete on ternary 3-caterpillars and on binary 4-caterpillars, where both the maximum degree and the hair length are bounded by small constants. Nevertheless, our linear-time algorithm for both problems on general 1-caterpillars, with unbounded degree, generalises the class of paths and cycles, and beats the complexity of the previous best known algorithm for paths. An interesting question is to answer whether the problems remain NP-complete on binary 3-caterpillars and on 2-caterpillars with bounded degree.

We also prove that COLOURFUL COMPONENTS is NP-complete on 5-coloured planar graphs with maximum degree 4 and on 12-coloured planar graphs with

maximum degree 3. A natural question is to ask whether the problem remains NP-complete when the number of colours is decreased but the maximum degree is 3 or 4.

References

1. Adamaszek, A., Popa, A.: Algorithmic and hardness results for the colorful components problems. Algorithmica **73**(2), 371–388 (2015)
2. Avidor, A., Langberg, M.: The multi-multiway cut problem. Theoret. Comput. Sci. **377**(1–3), 35–42 (2007)
3. Bousquet, N., Daligault, J., Thomassé, S.: Multicut is FPT. In: Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing, pp. 459–468. ACM (2011)
4. Bruckner, S., Hüffner, F., Komusiewicz, C., Niedermeier, R., Thiel, S., Uhlmann, J.: Partitioning into colorful components by minimum edge deletions. In: Kärkkäinen, J., Stoye, J. (eds.) CPM 2012. LNCS, vol. 7354, pp. 56–69. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-31265-6_5
5. Bulteau, L., Dabrowski, K.K., Fertin, G., Johnson, M., Paulusma, D., Vialette, S.: Finding a small number of colourful components. arXiv preprint arXiv:1808.03561 (2018)
6. Dondi, R., Sikora, F.: Parameterized complexity and approximation issues for the colorful components problems. Theoret. Comput. Sci. **739**, 1–12 (2018). https://doi.org/10.1016/j.tcs.2018.04.044
7. Fellows, M.R., Fertin, G., Hermelin, D., Vialette, S.: Upper and lower bounds for finding connected motifs in vertex-colored graphs. J. Comput. Syst. Sci. **77**(4), 799–811 (2011)
8. He, G., Liu, J., Zhao, C.: Approximation algorithms for some graph partitioning problems. In: Graph Algorithms and Applications, vol. 2, pp. 21–31. World Scientific (2004)
9. Hsu, W.L., Tsai, K.H.: Linear time algorithms on circular-arc graphs. Inf. Process. Lett. **40**(3), 123–129 (1991)
10. Hung, L.T.Q., Sysło, M.M., Weaver, M.L., West, D.B.: Bandwidth and density for block graphs. Discret. Math. **189**(1–3), 163–176 (1998)
11. Lichtenstein, D.: Planar formulae and their uses. SIAM J. Comput. **11**(2), 329–343 (1982)
12. Marx, D., Razgon, I.: Fixed-parameter tractability of multicut parameterized by the size of the cutset. SIAM J. Comput. **43**(2), 355–388 (2014)
13. Misra, N.: On the parameterized complexity of colorful components and related problems. In: Iliopoulos, C., Leong, H.W., Sung, W.-K. (eds.) IWOCA 2018. LNCS, vol. 10979, pp. 237–249. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-94667-2_20
14. Tovey, C.A.: A simplified NP-complete satisfiability problem. Discret. Appl. Math. **8**(1), 85–89 (1984)
15. Zheng, C., Swenson, K., Lyons, E., Sankoff, D.: OMG! orthologs in multiple genomes – competing graph-theoretical formulations. In: Przytycka, T.M., Sagot, M.-F. (eds.) WABI 2011. LNCS, vol. 6833, pp. 364–375. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-23038-7_30