On Intervalizing k-Colored Graphs for DNA Physical Mapping*

Hans L. Bodlaender§ Babette de Fluiter¶
Department of Computer Science, Utrecht University,
P.O. Box 80.089, 3508 TB Utrecht, the Netherlands

Abstract

The problem to determine whether a given k-colored graph is a subgraph of a properly colored interval graph has an application in DNA physical mapping. In this paper, we study the problem for the case that the number of colors k is fixed. For $k = 2$, we give a simple linear time algorithm, for $k = 3$, we give an $O(n^2)$ algorithm for biconnected graphs with n vertices, and for $k = 4$, we show that the problem is NP-complete.

1 Introduction

In this paper, we consider the following graph problem.

Intervalizing Colored Graphs [ICG]
Instance: A graph $G = (V, E)$, a coloring $c : V \to \{1,...,k\}$
Question: Is there a properly colored supergraph $G' = (V, E')$ of G which is an interval graph?

This problem models a problem arising in sequence reconstruction, which appears in some investigations in molecular biology (such as protein sequencing, nucleotide sequencing and gene sequencing, see [FHW93]). A sequence X (usually a large piece of DNA) is fragmented (or k copies of the sequence X are fragmented) such that the fragments can be further analyzed. The information about the order of the fragments in the original sequence is lost during the fragmentation process. The objective of DNA physical mapping is to reconstruct this order. To this end, a set of characteristics is

*This research was partially supported by the Foundation for Computer Science (S.I.O.N) of the Netherlands Organization for Scientific Research (N.W.O.) and partially by the ESPRIT Basic Research Actions of the EC under contract 7141 (project ALCOM II). An extended version of this paper, which also discusses ICG for general three-colored graphs, has appeared as [BdF95], and can be obtained with anonymous ftp from ftp.cs.ruu.nl file pub/RUU/CS/techreps/CS-1995/1995-15.ps.gz, or from the World Wide Web, URL http://www.cs.ruu.nl/~babette/research.html.

§E-mail: hansb@cs.ruu.nl
¶E-mail: babette@cs.ruu.nl

determined for each fragment (its 'fingerprint' or 'signature'), and based on respective fingerprints, an 'overlap' measure is computed. Using this overlap information, the fragments are assembled into islands of contiguous fragments (contigs).

Instances of ICG model the situation where k copies of X are fragmented, and some fragments (clones) are known to overlap. Fragments of the same copy of X will not overlap. Now each vertex in V represents one fragment; the color of a vertex represents to which copy of X the fragment belongs. It can be seen that ICG (and especially the constructive version of ICG, which also outputs an interval model of the interval graph G') helps here to predict other overlaps and to work towards reconstruction of the sequence X.

It has been known that ICG for an arbitrary number of colors is NP-complete [FHW93]. However, from the application it appears that the cases where the number of colors k (= the number of copies of X that are fragmented) is some small given constant are of interest. In this paper, we resolve the complexity of this problem for all constant values k. We observe that the case $k = 2$ is easy to resolve in linear time. Then, we give an $O(n^2)$ algorithm, that solves ICG for *biconnected* three-colored graphs. We also show how the algorithm can be made constructive. ICG can also be solved in $O(n^2)$ time for arbitrary three-colored graphs [BdF95], but an expose of this algorithm, while extending the methods used in this paper for biconnected graphs, would take too much space here. Finally, we show that ICG is NP-complete for four colors (and hence, for any fixed number of colors ≥ 4.)

In [FHW93], Fellows et al. consider ICG with a bounded number of colors. They show that, although for fixed $k \geq 3$, yes-instances have bounded pathwidth (and hence bounded treewidth), standard methods for graphs with bounded treewidth will be insufficient to solve ICG, as the problem is 'not finite state'. Also, they show ICG to be hard for the complexity class $W[1]$, (which was strengthened in [BFH94] to hardness for all classes $W[t]$, $t \in \mathbf{N}$). This result implies that it is unlikely that there exists a C, such that for any fixed number of colors k, ICG is solvable in time $O(f(k)n^C)$. Clearly, our NP-completeness result implies the fixed parameter intractability results, but is much stronger.

ICG is closely related to Triangulating Colored Graphs (TCG) where we look for a properly colored *triangulated* supergraph G' of a k-colored input graph G (i.e., G' does not contain a chordless cycle of length at least four). This problem is known to be NP-complete [BFW92], solvable in $O(n^{k+1})$ time for fixed k [MWW94], and solvable in linear time for the cases $k = 2$ and $k = 3$ [BK93, IS93, KW92, NON94]. Despite the close relationship between ICG and TCG, it appears that ICG poses some additional difficulties which require more complex and time consuming algorithms. For instance, while there is an easy characterization which assures that three-colored simple cycles can be triangulated without adding edges between vertices of the same color, for ICG on three-colored simple cycles, such a simple characterization does not exist, and even this case seems to require an $O(n^2)$ algorithm, based on dynamic programming. Additionally, TCG with three colors is 'finite state', while ICG with three colors is not (see [BFW92, FHW93]).

A generalization of ICG is INTERVALIZING SANDWICH GRAPHS (ISG). In this problem, the input is a *sandwich graph* $S = (V, E_1, E_2)$, where V is a set of vertices, and E_1 and E_2 are sets of edges between vertices of V, $E_1 \subseteq E_2$. The question is whether there is a graph $G = (V, E)$ such that G is an interval graph, and $E_1 \subseteq E \subseteq E_2$, i.e. E is 'sandwiched' between E_1 and E_2. This problem is NP-complete [GKS94], which also follows from NP-completeness of ICG. Our NP-completeness result for ICG with four colors also implies NP-completeness for ISG in which the clique size of the interval graph may be at most four.

Another closely related problem is UNIT-INTERVALIZING SANDWICH GRAPHS, which asks whether, for a given sandwich graph $S = (V, E_1, E_2)$, there is a graph $G = (V, E)$ such that G is a unit interval graph and $E_1 \subseteq E \subseteq E_2$. In [KS93, KST94], it is shown that this problem is NP-complete, polynomial for a fixed maximum clique size of G, hard for $W[1]$, but solvable in $O(n^{k-1})$ time if k is the maximum clique size of G.

This paper is organized as follows. In Section 2, necessary preliminary definitions and results are given, and the linear time algorithm for ICG on two-colored graphs is shown. A necessary condition for a three-colored graph G to be 'intervalizable' is that the pathwidth of G is at most two, or in other words, that G is a partial two-path [FHW93]. Hence, in Section 3, we analyze the structure of biconnected partial two-paths. In Section 4, we give our main algorithm, which is based on the analysis of Section 3 and dynamic programming, and in Section 5, we discuss our NP-completeness result.

2 Preliminaries

A graph G is a pair (V, E), where V is the set of vertices, and E is the set of edges. An edge is a set of two distinct vertices. The vertices and edges of a graph G are also denoted by $V(G)$ and $E(G)$, respectively.

Let G be a graph, $V' \subseteq V(G)$. The subgraph of G *induced* by V' is denoted by $G[V']$ and is defined as follows. $V(G[V']) = V'$ and $E(G[V']) = \{ e \in E(G) \mid e \subseteq V' \}$.

A *path* P in G is a sequence $(v_1, ..., v_s)$ of distinct vertices of G, such that there exists an edge between each pair of consecutive vertices.

A *cycle* is a graph C which consists of a path P containing all vertices of C, and an edge between the first and the last vertex of the path.

A *chordless* cycle C in G is a subgraph of G which is a cycle in which each two vertices which are not adjacent in C are also not adjacent in G.

A *biconnected* graph is a graph which remains connected if an arbitrary vertex is removed. A *biconnected component* B of a graph G is an induced subgraph of G which is biconnected and which is not a proper subgraph of another induced subgraph of G for which this holds. We only consider biconnected graphs and biconnected components which are non-trivial, i.e. which have at least three vertices.

A *tree* is a connected graph which contains no cycles. We usually denote trees by H instead of G.

An *interval graph* is a graph $G = (V, E)$ for which there is a function $\Phi : V \to \mathcal{I}$, where $\mathcal{I}$ is the set of all intervals on the real line, such that for each pair $v, w \in V$, $\Phi(v) \cap \Phi(w) \neq \emptyset \Leftrightarrow \{v, w\} \in E$. A k-coloring of a graph $G = (V, E)$ is a surjection $c : V \to \{1,\ldots,k\}$. A *proper* k-coloring is a k-coloring c such that for each edge $\{v, w\} \in E$, $c(v) \neq c(w)$. An *intervalization* of a graph $G = (V, E)$ with a k-coloring c, is a supergraph $G' = (V, E')$ of G ($E \subseteq E'$) which is an interval graph and is properly colored by c.

A *path decomposition* PD of a graph $G = (V, E)$ is a sequence $(V_1,\ldots,V_t)$, in which for all i, $V_i \subseteq V$ and V_i is non-empty, and the following conditions are satisfied:

1. For each $v \in V$, there is an i such that $v \in V_i$.
2. For each $e \in E$, there is an i such that $e \subseteq V_i$.
3. For each $i \leq j \leq l$, $V_i \cap V_l \subseteq V_j$.

The sets V_i are called the *nodes* of the path decomposition. The *width* of PD is $\max_i |V_i| \Leftrightarrow 1$. A graph G has *pathwidth* k if there is path decomposition of width k of G, but there is no path decomposition of width $k \Leftrightarrow 1$ or less of G. A graph G is called a *partial k-path* if it has pathwidth at most k.

Let G be a graph, $PD = (V_1,\ldots,V_t)$ a path decomposition of G. Let G' be a subgraph of G. The *occurrence* of G' in PD is a subsequence $(V_j,\ldots,V_{j'})$ of PD in which V_j and $V_{j'}$ contain an edge of G', and no node V_i, with $i < j$ or $i > j'$ contains an edge of G', i.e. $(V_j,\ldots,V_{j'})$ is the shortest subsequence of PD that contains all nodes of PD which contain an edge of G'. We say that G' *occurs* in $(V_j,\ldots,V_{j'})$. The vertices of G' occur in $(V_l,\ldots,V_{l'})$ if these are the only nodes in PD containing vertices of G'. An edge e is an *end edge* of G' if in each path decomposition of width two of G, e occurs in the left or right end node of the occurrence of G'. An edge $e \in E'$ is a *middle edge* of G' if in each path decomposition $PD = (V_1,\ldots,V_t)$ of width two of G in which G' occurs in $(V_j,\ldots,V_{j'})$, either $e \subseteq V_j$ or $e \subseteq V_{j'}$ or there is an i, $j \leq i \leq j'$, such that either $V_i \cap V(G') = e$ or $PD' = (V_1,\ldots,V_i,V_{i'},V_{i+1},\ldots,V_t)$ is a path decomposition of G and $V_{i'} \cap V(G') = e$.

Let G be a graph, $PD = (V_1,\ldots,V_t)$ a path decomposition of G. Let $1 \leq j \leq t$. We say that a node V_i is on the *left side* of V_j if $i < j$, and on the *right side* of V_j if $i > j$. Let G' be a connected subgraph of G, suppose G' occurs in $(V_l,\ldots,V_{l'})$. We say that G' occurs on the left side of V_j if $l' < j$, and on the right side of V_j if $l > j$. In the same way, we speak about the left and right sides of a sequence $(V_j,\ldots,V_{j'})$, i.e. a node is on the left side of $(V_j,\ldots,V_{j'})$ if it is on the left side of V_j, and a node is on the right side of $(V_j,\ldots,V_{j'})$ if it is on the right side of $V_{j'}$.

Let G be a graph, $PD = (V_1,\ldots,V_t)$ a path decomposition of G, $V' \subseteq V$ and suppose $G[V']$ occurs in $(V_j,\ldots,V_{j'})$, $1 \leq j \leq j' \leq t$. The path decomposition of $G[V']$ induced by PD is denoted by $PD[V']$ and is obtained from the sequence $PD[V'] = (V_j \cap V',\ldots,V_{j'} \cap V')$ by deleting all empty nodes and all nodes $V_i \cap V'$, $j \leq i < j'$, for which $V_i \cap V' = V_{i+1} \cap V'$.

Let $PD' = (W_1,...,W_{t'})$ be another path decomposition. The *concatenation* of PD and PD' is denoted by $PD +\!\!+ PD'$ and is defined as follows.

$$PD +\!\!+ PD' = (V_1,...,V_t, W_1,...,W_{t'})$$

Lemma 2.1. *Let $G = (V, E)$ be a graph, $PD = (V_1,...,V_t)$ a path decomposition of G. Let $G' = (V, E')$ be a supergraph of G with*

$$E' = \{ \{v, v'\} \mid \exists_{1 \leq i \leq t}\ v, v' \in V_i \}.$$

The graph G' is an interval graph.

Proof. Let $\Phi : V \rightarrow \{1,...,n\}$ be defined as follows. For each $v \in V$, if v occurs in nodes $(V_j,...,V_l)$, then take $\Phi(v) = [j, l]$. Then $\{v, v'\} \in E'$ if and only if $\Phi(v)$ and $\Phi(v')$ overlap. □

The graph G' is called the *interval completion* of G for PD.

A path decomposition $PD = (V_1,...,V_t)$ of a graph G which is k-colored is called a *proper path decomposition* if for each node V_i and each pair $v, w \in V_i$, if $v \neq w$ then $c(v) \neq c(w)$.

Lemma 2.2. *Let $G = (V, E)$ be a graph, $c : V \rightarrow \{1,...,k\}$ a k-coloring of G. G has an intervalization if and only if there is a proper path decomposition of G, which has width at most $k \Leftrightarrow 1$.*

Proof. (See also [FHW93].) For the 'if' part, suppose $PD = (V_1,...,V_t)$ is a proper path decomposition of G. Note that PD has width $k \Leftrightarrow 1$. Then the interval completion of G for PD is a properly k-colored interval graph.

For the 'only if' part, suppose $G' = (V, E')$ is an intervalization of G. Let $\Phi : V \rightarrow \mathcal{I}$ be a function for G' such that for each $v, w \in V$, $v \neq w$, $\{v, w\} \in E \Leftrightarrow \Phi(v) \cap \Phi(w) \neq \emptyset$. Suppose w.l.o.g. that for each $v \in V$, $\Phi(v)$ is a closed interval. For each $v \in V$, let $l(v)$ denote the leftmost element in $\Phi(v)$. Let $(u_1,...,u_n)$, $n = |V|$, be an ordering of V such that for each $i < j$, $l(u_i) \leq l(u_j)$. For each i let $V_i = \{ v \in V \mid l(u_i) \in \Phi(v) \}$. Then $PD = (V_1,...,V_n)$ is a proper path decomposition of G' and hence of G. Furthermore, each node contains at most k vertices, since there are at most k vertices with different colors. Hence PD has pathwidth at most $k \Leftrightarrow 1$. □

Thus, the following problem is equivalent to ICG.

Proper Path Decomposition [PPD]
Instance: A graph $G = (V, E)$, a k-coloring $c : V \rightarrow \{1,...,k\}$
Question: Is there a proper path decomposition of G?

In this paper, we use both ICG and PPD. Note that the proof of Lemma 2.2 also gives an easy way to transform a solution for one problem into a solution for the other problem.

For the case that $k = 2$, the question whether there is a proper path decomposition of G is equal to the question whether G is a properly colored partial one-path (see also [FHW93]). This is because if G is properly colored, then we can transform each path decomposition of width one of G into a proper path decomposition of width one by simply deleting all nodes which contain no edge, and then adding a node at the right side of the path decomposition for each isolated vertex containing this vertex only. Checking whether a graph has pathwidth one can be done in linear time, and checking whether it is properly colored also.

Theorem 2.1. *For $k = 2$, ICG can be solved in linear time.*

We now give some lemmas, which are frequently used in the remainder of this report.

The following two lemmas are well-known.

Lemma 2.3. *Let $(V_1,...,V_r)$ be a path decomposition of $G = (V, E)$. Suppose $i < j < k$, and suppose P is a path from $v \in V$ to $w \in V$, $v \in V_i$, $w \in V_k$. Then V_j contains at least one vertex from P.*

Proof. Follows from the definition of path decompositions by induction on the length of the path. □

The following Lemma is proved in e.g. [BM93].

Lemma 2.4. (*Clique Containment*) *Let $G = (V, E)$ be a graph, $PD = (V_1,...,V_t)$, a path decomposition of G, suppose $V' \subseteq V$ forms a clique in G. There is an i, $1 \leq i \leq t$, such that $V' \subseteq V_i$.*

Proof. We prove this by induction on $|V'|$. If $|V'| = 2$, then there is a V_i containing V' by definition. Suppose $|V'| > 2$. Let $v \in V'$. There is a node V_i, such that $V' \Leftrightarrow \{v\} \subseteq V_i$. Suppose v occurs in $(V_j,...,V_{j'})$. Suppose w.l.o.g. that $i \leq j'$. If $i \geq j$, then clearly $V' \subseteq V_i$. If $i < j$, then for each $w \in V'$, there is an l, $j \leq l \leq j'$, such that $w \in V_l$. Hence $V' \subseteq V_j$, which gives a contradiction. □

3 The Structure of Biconnected Partial Two-Paths

In this section, we give a characterization of biconnected partial two-paths.

Given a graph $G = (V, E)$, the graph $\bar{G}$ which is obtained from G by adding all edges $\{v, w\} \notin E$ such that there are three disjoint paths from v to w in G is called the *cell completion* of G. (Two paths from v to w are disjoint if they only have vertices v and w in common.) The following lemma has been proved in [BK93] in the setting of partial two-trees.

Lemma 3.1. *Let G be a partial two-path. The cell completion $\bar{G}$ of G is a subgraph of any intervalization of G of pathwidth at most two.*

In terms of path decomposition, the lemma states that each path decomposition of width two of a partial two-path G is a path decomposition of the cell completion $\bar{G}$. The cell completion of a partial two-path can be found in linear time [BK93]. In the cell completion of a graph, each two distinct chordless cycles have at most one edge in common. In [BK93], it has been shown that the cell completion of a biconnected partial two-tree is a tree of chordless cycles. We show that the cell completion of a biconnected partial two-path is a path of chordless cycles. Before we prove this, we first give definitions and prove a number of lemmas.

Definition 3.1. (Path of Chordless Cycles). *A* path of chordless cycles *is a pair $(\mathcal{C}, \mathcal{S})$, where $\mathcal{C}$ is a sequence $(C_1,\ldots,C_p)$ of chordless cycles, $p \geq 1$, and $\mathcal{S}$ is a sequence $(e_1,\ldots,e_{p-1})$ of edges, such that for each i and j, $1 \leq i < j \leq p$, $V(C_i) \cap V(C_j) = e_i \cap e_{j-1}$, $E(C_i) \cap E(C_j) = \{e_i\} \cap \{e_{j-1}\}$, and for each i, $1 \leq i < p \Leftrightarrow 1$, if $e_i = e_{i+1}$, then $|V(C_{i+1})| = 3$.*

In Figure 1, an example of a path of chordless cycles is given with six chordless cycles.

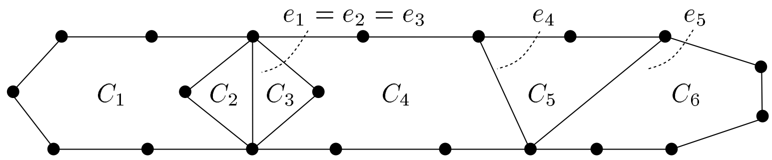

Figure 1: A path of chordless cycles $(\mathcal{C}, \mathcal{S})$ with $\mathcal{C} = (C_1,\ldots,C_6)$, $\mathcal{S} = (e_1,\ldots,e_5)$. $V(C_1) = \{1,2,3,4,16,17,18\}$, $V(C_2) = \{4,16,19\}$, $V(C_3) = \{4,16,20\}$, $V(C_4) = \{4,5,6,13,14,15,16\}$, $V(C_5) = \{6,7,8,13\}$ and $V(C_6) = \{8,9,10,11,12,13\}$. Furthermore, $e_1 = e_2 = e_3 = \{4,16\}$, $e_4 = \{6,13\}$ and $e_5 = \{8,13\}$.

Definition 3.2. *Let G be a biconnected graph, $(\mathcal{C}, \mathcal{S})$ a path of chordless cycles, where $\mathcal{C} = (C_1,\ldots,C_p)$, $\mathcal{S} = (e_1,\ldots,e_{p-1})$, $p \geq 1$. $(\mathcal{C}, \mathcal{S})$ is a path of chordless cycles for G if $V(G) = \bigcup_{i=1}^{p} V(C_i)$ and $E(G) = \bigcup_{i=1}^{p} E(C_i)$.*

Lemma 3.2. *Let G be a biconnected partial two-path, C a cycle of $\bar{G}$, and $PD = (V_1,\ldots,V_t)$ a path decomposition of G of width two. Suppose C occurs in $(V_j,\ldots,V_{j'})$, and $\{x,y\}$ is an edge of C occurring in V_j, $\{x',y'\}$ an edge occurring in $V_{j'}$. The following holds.*

1. *If $|V(C)| > 3$, then $\{x,y\} \neq \{x',y'\}$.*
2. *For each i, $j \leq i \leq j'$, $|V_i \cap V(C)| \geq 2$ and for each edge $e \in E(C)$ there is an i, $j \leq i \leq j'$, such that $e \subseteq V_i$ and $|V_i \cap V(C)| = 3$.*

Proof. 1. Suppose $x = x'$, $y = y'$. Because $|V(C)| > 3$, there is an edge $\{v,w\}$ in C with $\{v,w\} \cap \{x,y\} = \emptyset$. Because of the definition of path decomposition and Lemma 2.3, there must be a V_i, $j \leq i \leq j'$, with $v,w,x,y \in V_i$, hence $|V_i| \geq 4$.

2. Suppose w.l.o.g. that x and x' are connected by a path in C which does not contain y or y'. Denote this path by P_1. Denote the path between y and y' not containing x or x' by P_2. See also Figure 2. According to Lemma 2.3, each V_i, $j \leq i \leq j'$, contains a vertex of P_1. Analogously, each V_i contains a vertex of P_2. Since P_1 and P_2 are vertex disjoint, $|V_i \cap V(C)| \geq 2$ for each i, $j \leq i \leq j'$. Suppose P_1 contains at least one edge. Let e be an edge of P_1. Let V_l, $j \leq l \leq j'$ such that $e \subseteq V_l$. This V_l also contains a vertex of P_2, hence there is an i such that $e \subseteq V_i$ and $|V_i \cap V(C)| \geq 3$ for each edge e on P_1 and P_2. Now consider edge $\{x, y\} \subseteq V_j$. If there is another vertex of C in V_j, then the lemma holds for $\{x, y\}$. If $V_j \cap V(C) = \{x, y\}$, then there must be an i, $j \leq i \leq j'$, such that $\{x, y\} \subseteq V_i$ and V_i contains a neighbor of x or y. Hence $|V_i \cap V(C)| = 3$. Similar for edge $\{x', y'\}$. □

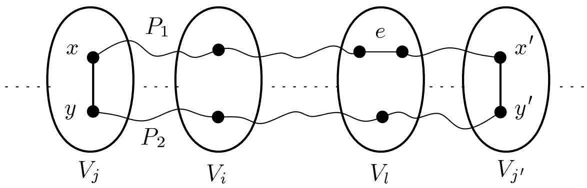

Figure 2: The occurrence of chordless cycle C as in part 2 of the proof of Lemma 3.2.

Let G be a biconnected partial two-path. Lemma 3.2 implies that the occurrences of two chordless cycles of $\bar{G}$ which do not have a vertex in common can not overlap in any path decomposition of width two of G. If two chordless cycles have one edge in common, then the occurrences of these two cycles can only overlap in their common edge, as we show in the next lemma.

Lemma 3.3. *Let G be a biconnected partial two-path with cycles C and C' which have one edge $\{x, y\}$ and no other vertices in common. Let $PD = (V_1, ..., V_t)$ be a path decomposition of G of pathwidth two. Suppose C occurs in $(V_j, ..., V_{j'})$, C' occurs in $(V_l, ..., V_{l'})$. Then the following holds.*

1. *$j \leq l$ and $j' \leq l'$ or $j \geq l$ and $j' \geq l'$. If $j = l$ and $j' = l'$, then $|V(C)| = |V(C')| = 3$.*

2. *If $j \leq l$, $j' \leq l'$, then $j' \geq l$, $\{x, y\}$ is an end edge of C and of C' and it occurs in $V_{j'}$ and in V_l, and there is an i, $l \leq i < j'$, such that $V(C) \cap (V_{i+1} \cup ... \cup V_t) = \{x, y\}$ and $V(C') \cap (V_1 \cup ... \cup V_i) = \{x, y\}$ (or possibly vice versa, if $j = l$ and $j' = l'$), so $\{x, y\}$ is a middle edge of $C \cup C'$.*

Proof. 1. Suppose $j < l$ and $j' > l'$, then $|V(C')| = 3$, say $V(C') = \{x, y, z\}$, since each of $V_j, ..., V_{j'}$ contains two vertices of C. Let $j < i < j'$, such that $V_i = \{x, y, z\}$. Suppose $\{a, b\}, \{c, d\} \in E(C)$ and $\{a, b\} \subseteq V_j$, $\{c, d\} \subseteq V_{j'}$, such that there is a path from a to c not containing b or d. Let P_1 denote this path, and P_2 denote the path from b to d not containing a and c. $\{a, b\} \neq \{x, y\}$ and $\{c, d\} \neq \{x, y\}$, so suppose

$\{x,y\} \in E(P_1)$. V_i contains a vertex of P_2, which is not x, y or z. Hence $|V_i| \geq 4$, which is a contradiction. So either $j \leq l$ and $j' \leq l'$ or $j \geq l$ and $j' \geq l'$. If $j = l$ and $j' = l'$, then $|V(C)| = |V(C')| = 3$, since each V_i, $j \leq i \leq j'$, contains two vertices of C and two vertices of C'.

2. It is clear that $j' \geq l$, since $\{x,y\}$ is an edge of both C and C'. There are nodes V_m and $V_{m'}$ such that $V_m = \{x,y,z\}$ for some $z \in V(C)$ with $z \neq x,y$, and $V_{m'} = \{x,y,z'\}$ for some $z' \in V(C')$ with $z' \neq x,y$. Note that $l \leq m, m' \leq j'$. Suppose first that $l \leq m < m' \leq j'$. We show that all vertices of $V(C) \Leftrightarrow \{x,y\}$ occur only on the left side of $V_{m'}$. Suppose there is a vertex $v \in V(C) \Leftrightarrow \{x,y\}$ which occurs on the right side of $V_{m'}$. There is a path from v to z in C which does not contain x and y. Node $V_{m'}$ contains a vertex of this path. Hence $|V_{m'}| \geq 4$. This is a contradiction. Since each V_i, $m \leq i \leq m'$, contains x and y, this means that there is an i, $m \leq i < m'$, such that all vertices of $V(C) \Leftrightarrow \{x,y\}$ occur only in $(V_1,...,V_i)$, and the vertices of $V(C') \Leftrightarrow \{x,y\}$ occur only in $(V_{i+1},...,V_t)$. Furthermore, since $i < j'$ and $V_{j'}$ contains an edge of C, $V_{j'}$ contains x and y. Similarly, V_l contains x and y.

Now suppose $l \leq m' < m \leq j'$. In the same way as before, we can show that the vertices of $V(C) \Leftrightarrow \{x,y\}$ occur only on the right side of $V_{m'}$, and the vertices of $V(C') \Leftrightarrow \{x,y\}$ occur only on the left side of V_m. Hence there is an i, $m' \leq i < m$, such that all vertices of $V(C) \Leftrightarrow \{x,y\}$ occur only in $(V_{i+1},...,V_t)$ and all vertices of $V(C') \Leftrightarrow \{x,y\}$ occur only in $(V_1,...,V_i)$. Furthermore, V_l is the leftmost node which contains an edge of C', which means that $j = l$. In the same way, we can prove that $j' = l'$, and V_l and $V_{j'}$ both contain x and y. □

Note that in part 2 of the lemma, the part $(V_j,...,V_i)$ of PD restricted to $V(C)$ is a path decomposition of C, and $(V_{i+1},...,V_l)$ restricted to $V(C')$ is a path decomposition of C'. We say that C occurs on the left side of C'. In other words, Lemma 3.3 says that, if there are two cycles which have one edge in common, then in each path decomposition, one occurs on the left side of the other one.

Lemma 3.4. *Let G be a biconnected partial two-path, C a chordless cycle of $\bar{G}$ which has edges e_1 and e_2, $e_1 \neq e_2$, in common with chordless cycles C_1 and C_2, respectively. If C_1 and C_2 have a vertex v in common, then $v \in e_1 \cap e_2$.*

Proof. Suppose C_1 and C_2 have vertex v in common, and $v \notin e_1 \cap e_2$. It can be shown that for each pair of vertices $u, u' \in e_1 \cup e_2 \cup \{v\}$, $u \neq u'$, there are three or more disjoint paths between u and u' in G. Hence there is an edge between u and u' in $\bar{G}$, so the vertices of $e_1 \cup e_2 \cup \{v\}$ form a clique in $\bar{G}$. But $|e_1 \cup e_2 \cup \{v\}| \geq 4$, which means that, according to Lemma 2.4, $\bar{G}$ has pathwidth three or more. □

Lemma 3.5. *Let G be a biconnected partial two-path, C a chordless cycle in $\bar{G}$. If C has two distinct edges in common with two other chordless cycles C_1 and C_2 of $\bar{G}$, then C_1 and C_2 can not both occur on the same side of the occurrence of C.*

Proof. Let $PD = (V_1,...,V_t)$ be a path decomposition of width two of G, suppose C occurs in $(V_j,...,V_{j'})$. Suppose $e_1 = \{x_1,y_1\}$ and $e_2 = \{x_2,y_2\}$ are the edges that

C has in common with C_1 and C_2, respectively, and C_1 and C_2 occur on the left side of C. Then e_1 and e_2 occur in V_j. e_1 and e_2 must have a common vertex, otherwise $|V_j| \geq 4$, say $y_1 = x_2$. All vertices of C_1 and C_2 other than x_1, x_2 and y_2 occur only on the left side of V_j, since V_j contains x_1, x_2 and y_2 (see proof of Lemma 3.3). Suppose the leftmost edge of C_1 occurs in V_l, the leftmost edge of C_2 occurs in $V_{l'}$, and $l \leq l'$. Then each V_i, $l' \leq i \leq j$, contains at least two vertices of C_1 and there is a V_i which contains three vertices of C_2. Because of Lemma 3.4, C_1 and C_2 have only one vertex in common, which means that $|V_i| \geq 4$. □

The following corollary follows directly from Lemma 3.5.

Corollary 3.1. *Let G be a biconnected partial two-path, C a chordless cycle in $\bar{G}$. C has at most two edges in common with two other chordless cycles.*

We have now shown that the chordless cycles of the cell completion of a biconnected partial two-path form a sequence, such that each chordless cycle has exactly one edge in common with the following chordless cycle in the sequence.

Lemma 3.6. *Let G be a partial two-path, let $e \in E(G)$ such that e is an edge of three or more chordless cycles of $\bar{G}$. There are at most two chordless cycles which contain e and have four or more vertices, and in each path decomposition of width two of $\bar{G}$, these chordless cycles occur leftmost or rightmost of all chordless cycles containing e.*

Proof. Suppose e is an edge of $s \geq 3$ chordless cycles C_i, $3 \leq i \leq s$. Let PD be a path decomposition of width two of G, and suppose w.l.o.g. that C_i occurs on the left side of C_j for all i and j with $i < j$. Since C_1 and C_s have x and y in common, x and y occur in the first and the last V_j containing an edge of all C_i with $1 < i < s$. Hence $|V(C_i)| = 3$ for all i, $1 < i < s$, so only C_1 and C_s may have four or more vertices. □

We can now prove the main result of this section.

Theorem 3.1. *Let G be a biconnected graph. G is a partial two-path if and only if there is a path of chordless cycles for $\bar{G}$.*

Proof. Suppose $(\mathcal{C}, \mathcal{S})$ is a path of chordless cycles for $\bar{G}$, with $\mathcal{C} = (C_1, \ldots, C_p)$ $\mathcal{S} = (e_1, \ldots, e_{p-1})$, $p \geq 1$. Then we can make a path decomposition of width two of G as follows. Let e_0 be an arbitrary edge in C_1 with $e_0 \neq e_1$, and let e_p be an arbitrary edge in C_p with $e_p \neq e_{p-1}$. For each i, $1 \leq i \leq s$, we make a path decomposition PD_i of C_i as follows. If $|V(C_i)| = 3$, let $PD_i = (V(C_i))$. Otherwise, do the following. Let $e_{i-1} = \{x, y\}$ and $e_i = \{x', y'\}$ such that there is a path from x to x' which does not contain y or y', Let $P_1 = (u_1, \ldots, u_q)$ denote the path in C_i from x to x' which does not contain y or y', and let $P_2 = (v_1, \ldots, v_r)$ denote the path in C_i from y to y' not containing x or x'. For each j, $1 \leq j < q$, let $V_j = \{u_j, u_{j+1}, v_1\}$, and for each j, $1 \leq j < r$, let $V_{j+q-1} = \{u_q, v_j, v_{j+1}\}$. Let $PD_i = (V_1, \ldots, V_{q+r-2})$. Note that $e_{i-1} \subseteq V_1$ and $e_i \subseteq V_{q+r-2}$. $PD = PD_1 \mathbin{+\!\!+} PD_2 \mathbin{+\!\!+} \cdots \mathbin{+\!\!+} PD_p$ is a path decomposition of width two of G.

In Figure 3, an example of a path decomposition of width two is given for the graph of Figure 1.

If G is a partial two-path, then Lemmas 3.1 up to 3.6 show that $\bar{G}$ can be written as a path of chordless cycles. □

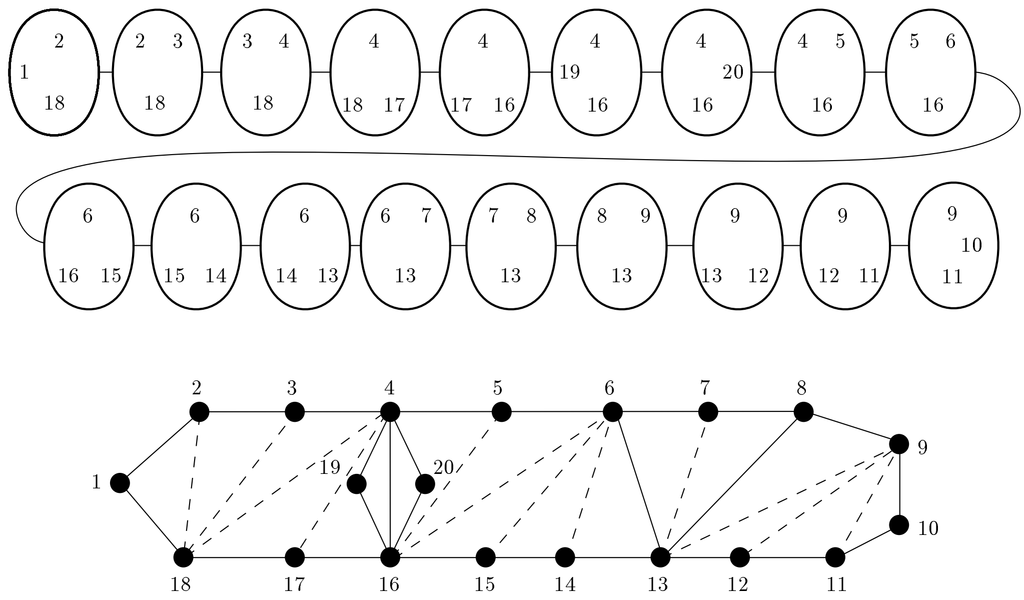

Figure 3: A path decomposition of width two for the graph of Figure 1 and its corresponding interval completion. The path decomposition is constructed as in the proof of Theorem 3.1, with $e_0 = \{1, 18\}$ and $e_p = \{9, 10\}$. The dashed edges in the graph are the edges that are added for the interval completion.

In the same way as in [BK93], we can check whether $\bar{G}$ is a tree of chordless cycles, and make a list of all chordless cycles in linear time. After that, we can check in linear time whether the tree of chordless cycles is a path of chordless cycles.

4 Intervalizing Biconnected Three-Colored Graphs

In this section, we give an algorithm for determining whether there is an intervalization of a given biconnected, three-colored graph. The main algorithm has the following form: first, a path of chordless cycles for $\bar{G}$ is constructed if it exists (see Section 3). Then this path of chordless cycles is used to check if there is a proper path decomposition of G.

The following lemma follows directly from the lemmas and theorem of Section 3.

Lemma 4.1. *Let G be a biconnected partial two-path, $(\mathcal{C}, \mathcal{S})$ a path of chordless cycles of $\bar{G}$ with $\mathcal{C} = (C_1, ..., C_p)$ and $\mathcal{S} = (e_1, ..., e_{p-1})$. There is a nice proper path decomposition of $\bar{G}$ if and only if the following holds:*

1. *there is a proper path decomposition of* C_1 *with edge* e_1 *in the rightmost node,*

2. *there is a proper path decomposition of* C_p *with edge* e_{p-1} *in the leftmost node, and*

3. *for all* i, $1 < i < p$, *there is a proper path decomposition of* C_i *with edge* e_{i-1} *in the leftmost node and edge* e_i *in the rightmost node.*

Hence to check whether there is a proper path decomposition of G, the algorithm can check for each chordless cycle C_i, $1 \leq i \leq p$, whether there is a proper path decomposition of C_i with the appropriate edges in the leftmost and the rightmost node. The proper path decompositions of the chordless cycles can then be concatenated in the order in which they occur in the path of chordless cycles of G, and this gives a proper path decomposition of G.

Hence we concentrate now on checking whether there exists a proper path decomposition of a chordless cycle C. Let C be a properly three-colored chordless cycle. We denote the vertices and edges of C by $V(C) = \{v_0, v_1, ..., v_{n-1}\}$, and $E(C) = \{\, \{v_i, v_{i+1}\} \mid 0 \leq i < n \,\}$ (for each i, let v_i denote $v_{i \bmod n}$). For each j and l, $1 \leq l < n$, let $I(j,l)$ denote the set of vertices of $V(C)$ between v_j and v_{j+l}, when going from v_j to v_{j+l} in positive direction, i.e.,

$$I(j,l) = \{\, v_i \mid j \leq i \leq j+l \,\}.$$

Furthermore, let $C(j,l)$ denote the cycle with

$$\begin{aligned} &V(C(j,l)) = I(j,l) \\ &E(C(j,l)) = \{\{v_j, v_{j+l}\}\} \cup \{\, \{v_i, v_{i+1}\} \mid v_i \in I(j,l) \Leftrightarrow \{v_{j+l}\} \,\} \end{aligned}$$

Note that $C(j, n \Leftrightarrow 1) = C$ for all j. The following lemma is used to obtain a dynamic programming algorithm for our problem.

Lemma 4.2. *Let* C *be a properly three-colored cycle. Let* i, j *and* l *be integers,* $2 \leq l < n$. *There is a proper path decomposition* $PD = (V_1, ..., V_t)$ *of* $C(j,l)$ *such that* $\{v_i, v_{i+1}\} \subseteq V_1$ *and* $\{v_j, v_{j+l}\} \subseteq V_t$ *if and only if* $c(v_j) \neq c(v_{j+l})$ *and either one of the following conditions holds:*

1. $|V(C)| = 3$,

2. *there is a proper path decomposition* $PD' = (V'_1, ..., V'_r)$ *of* $C(j, l \Leftrightarrow 1)$ *such that* $\{v_i, v_{i+1}\} \subseteq V'_1$ *and* $\{v_j, v_{j+l-1}\} \subseteq V'_r$, *or*

3. *there is a proper path decomposition* $PD'' = (V''_1, ..., V''_s)$ *of* $C(j+1, l \Leftrightarrow 1)$ *such that* $\{v_i, v_{i+1}\} \subseteq V''_1$ *and* $\{v_{j+1}, v_{j+l}\} \subseteq V''_s$.

Proof. For the 'if' part, suppose $c(v_j) \neq c(v_{j+l})$. If $|V(C)| = 3$, then $C(j,l) = C$, and hence $(V(C))$ is a proper path decomposition of C, since C is properly colored. Suppose there is a proper path decomposition $PD' = (V'_1, ..., V'_r)$ of $C(j, l \Leftrightarrow 1)$ with

$\{v_i, v_{i+1}\} \subseteq V'_1$ and $\{v_j, v_{j+l-1}\} \subseteq V'_r$. Then $PD = PD' \mathbin{+\!\!+} (\{v_j, v_{j+l-1}, v_{j+l}\})$ is a proper path decomposition of $C(j,l)$ which satisfies the appropriate conditions. The other case is similar.

For the 'only if' part, suppose there is a proper path decomposition $PD = (V_1,..., V_t)$ of C such that $\{v_i, v_{i+1}\} \subseteq V_1$ and $\{v_j, v_{j+l}\} \subseteq V_t$. Clearly, $c(v_j) \neq c(v_{j+l})$, since $v_j, v_{j+l} \in V_t$. Suppose $|V(C)| > 3$. If $\{v_i, v_{i+1}\} = \{v_j, v_{j+l}\}$, then $|C(j,l)| = |V(C)| > 3$, and Lemma 3.2 shows that the leftmost and the rightmost node of PD can not contain the same edge, contradiction. So $\{v_i, v_{i+1}\} \neq \{v_j, v_{j+l}\}$. Let V_m and $V_{m'}$, $1 \leq m, m' \leq t$, be the rightmost nodes containing edge $\{v_{j+1}, v_j\}$ and $\{v_{j+l-1}, v_{j+l}\}$, respectively.

First suppose $m' < m$. Then $V_m = \{v_{j+1}, v_j, v_{j+l}\}$. Furthermore, for each k, $m < k \leq t$, $V_k = \{v_j, v_{j+l}\}$, since if there is a V_k, $m < k \leq t$, such that $v \in V_k$ for some $v \in V(C) \Leftrightarrow \{v_j, v_{j+l}\}$, then $v \in V_m$, which gives a contradiction. Note that $v_j \notin V_{m'}$, since $V_{m'}$ contains v_{j+l}, v_{j+l-1}, and a vertex of the path from v_{j+1} to v_{j+l-1} which does not contain v_j. Hence $v_j \notin V_1$. Let PD' be the path decomposition obtained from $(V_1,..., V_m)$ by deleting v_j from all nodes containing it. Then PD' is a proper path decomposition of $C(j+1, l \Leftrightarrow 1)$ with edge $\{v_{j+1}, v_{j+l}\}$ in the rightmost node and edge $\{v_i, v_{i+1}\}$ in the leftmost node.

For the case that $m < m'$, a proper path decomposition for $C(j, l \Leftrightarrow 1)$ with $\{v_i, v_{i+1}\}$ in the leftmost node and $\{v_j, v_{j+l-1}\}$ in the rightmost node can be constructed in the same way.

If $m = m'$, then $v_{j+1} = v_{j+l-1}$, hence $|V(C(j,l))| = 3$. Since $\{v_i, v_{i+1}\} \neq \{v_j, v_{j+l}\}$, this means that $\{v_i, v_{i+1}\} = \{v_j, v_{j+1}\}$ or $\{v_i, v_{i+1}\} = \{v_{j+l-1}, v_{j+l}\}$. In the first case, $(\{v_i, v_{i+1}\})$ is a proper path decomposition of $C(j, l \Leftrightarrow 1)$ with edge $\{v_i, v_{i+1}\}$ in the leftmost node and edge $\{v_j, v_{j+l-1}\}$ in the rightmost node. In the latter case, $(\{v_i, v_{i+1}\})$ is a proper path decomposition of $C(j+1, l \Leftrightarrow 1)$ with edge $\{v_i, v_{i+1}\}$ in the leftmost node and edge $\{v_{j+1}, v_{j+l}\}$ in the rightmost node. □

Let C be a chordless cycle of a properly three-colored path of chordless cycles. The set of edges of which one must occur in the leftmost end node of the proper path decomposition of C is called the set of *starting edges*, and is denoted by E_{S}. In the same way, the set of *ending edges* E_{E} is defined to be the set of edges of which one must occur in the rightmost end node of the proper path decomposition of C_i.

We define $PPW2$ as follows. Let $E_{\mathrm{S}} \subseteq E(C)$ be a set of starting edges, and let j and l be integers, $1 \leq l < n$.

$$PPW2(C, E_{\mathrm{S}}, j, l) = \begin{cases} \text{true} & \text{if } \exists_{PD=(V_1,..,V_t)}\, PD \text{ is a proper path decomposition} \\ & \text{of } C(j,l) \ \wedge v_j, v_{j+l} \in V_t \wedge \exists_{e \in E_{\mathrm{S}}}\, e \subseteq V_1 \\ \text{false} & \text{otherwise} \end{cases}$$

There is a proper path decomposition of C with an edge from E_{S} in the leftmost node and an edge from E_{E} in the rightmost node if and only if $PPW2(C, E_{\mathrm{S}}, j, n \Leftrightarrow 1)$ holds for some j for which $\{v_{j-1}, v_j\} \in E_{\mathrm{E}}$.

If $|V(C)| = 3$, then there is a proper path decomposition with an edge from E_{S} in the leftmost node and an edge from E_{E} in the rightmost node if and only if C is properly colored. This path decomposition can consist of one node, which contains $V(C)$.

Suppose $|V(C)| > 3$. We use Lemma 4.2 to describe $PPW2$ recursively. Let$E_{\mathrm{S}} \subseteq E(C)$, and let j and l be integers, $1 \leq l < n$.

$$PPW2(C, E_{\mathrm{S}}, j, l) = \begin{cases} \{v_j, v_{j+l}\} \in E_{\mathrm{S}} & \text{if } l = 1 \\ c(v_j) \neq c(v_{j+l}) \wedge & \\ (\, PPW2(C, E_{\mathrm{S}}, j+1, l \Leftrightarrow 1) \vee PPW2(C, E_{\mathrm{S}}, j, l \Leftrightarrow 1)\,) & \text{if } l > 1 \end{cases}$$

It can be seen from the definition of $PPW2$ that $PPW2(C, E_{\mathrm{S}}, j, 1)$ holds if and only if $\{v_j, v_{j+1}\} \in E_{\mathrm{S}}$.

For a given properly three-colored cycle C, $|V(C)| = n$, and set of starting edges $E_{\mathrm{S}} \subseteq E(C)$, and ending edges $E_{\mathrm{E}} \subseteq E(C)$, we can compute whether there is a proper path decomposition of C with these starting and ending edges in $O(n^2)$ time using dynamic programming with the following procedure.

Algorithm 1
Procedure COMP_PPW2$(n, C, c, E_{\mathrm{S}}, E_{\mathrm{E}})$
Input:
  Integer $n \geq 3$
  Cycle C with n vertices $v_0, ..., v_{n-1}$, edges $\{\,\{v_i, v_{i+1}\} \mid 0 \leq i < n\,\}$
  Proper three-coloring $c : V \rightarrow \{1, 2, 3\}$
  Set of starting edges $E_{\mathrm{S}} \subseteq E(C)$
  Set of ending edges $E_{\mathrm{E}} \subseteq E(C)$
Output: $\exists_{0 \leq j < n} \{v_{j-1}, v_j\} \in E_{\mathrm{E}} \wedge PPW2(C, E_{\mathrm{S}}, j, n \Leftrightarrow 1)$

if $n = 3$ **return** true
for $j \leftarrow 0$ **to** $n \Leftrightarrow 1$
  do $P(j, 1) \leftarrow$ false
for all $\{v_j, v_{j+1}\} \in E_{\mathrm{S}}$
  do $P(j, 1) \leftarrow$ true
$(* \; \forall_{0 \leq j < n} P(j, 1) \equiv PPW2(C, E_{\mathrm{S}}, j, 1) \; *)$
for $l \leftarrow 2$ **to** $n \Leftrightarrow 1$
  do for $j \leftarrow 0$ **to** $n \Leftrightarrow 1$
    do $P(j, l) \leftarrow (c(v_j) \neq c(v_{j+l})) \wedge (P((j+1) \bmod n, l \Leftrightarrow 1) \vee P(j, l \Leftrightarrow 1))$
$(* \; \forall_{0 \leq j < n} P(j, n \Leftrightarrow 1) \equiv PPW2(C, E_{\mathrm{S}}, j, n \Leftrightarrow 1) \; *)$
for all $\{v_j, v_{j-1}\} \in E_{\mathrm{E}}$
  do if $P(j, n \Leftrightarrow 1)$
    then return true
return false

Let G be a three-colored biconnected partial two-path, $(\mathcal{C}, \mathcal{S})$ a path of chordless cycles for G with $\mathcal{C} = (C_1, ..., C_p)$. There is a proper path decomposition of G if and only if for each i, $1 \leq i \leq p$, there is a proper path decomposition of C_i with set of starting edges $\{e_{i-1}\}$ if $i > 1$, $E(C_i)$ otherwise, and set of ending edges $\{e_i\}$ if $i < p$, $E(C_i)$ otherwise.

For a given three-colored biconnected graph G, the algorithm is now as follows.

Algorithm 2

1. Find the cell completion $\bar{G}$ of G and check if $\bar{G}$ is a tree of chordless cycles, and is properly three-colored. If not, stop, the answer is no.
2. Check if there is a path of chordless cycles for $\bar{G}$. If so, construct such a path $(\mathcal{C}, \mathcal{S})$ with $\mathcal{C} = (C_1, ..., C_p)$ and $\mathcal{S} = (e_1, ..., e_{p-1})$. If not, stop, the answer is no.
3. For each chordless cycle C_i in the path, let $m = |V(C_i)|$, let $E_{\mathrm{S}} = \{e_{i-1}\}$ if $i > 1$, otherwise $E_{\mathrm{S}} = E(C_i)$, and let $E_{\mathrm{E}} = \{e_{i+1}\}$ if $i < p$, $E_{\mathrm{E}} = E(C_i)$ otherwise. Compute COMP_PPW2$(m, C_i, c, E_{\mathrm{S}}, E_{\mathrm{E}})$. If the computed value is true for each C_i, the answer is yes, otherwise it is no.

Step 1 and 2 can be made to run in $O(n)$ time (see Section 3 and [BK93]). Step 3 runs in $O(n^2)$ time ($n = |V(G)|$) if procedure COMP_PPW2 is used. Hence, we have proved our main result here:

Theorem 4.1. *There exists an $O(n^2)$ time algorithm, that solves ICG for biconnected three-colored graphs.*

The algorithm can be made constructive, in the sense that if there exists an intervalization, then the algorithm outputs one, as follows. In procedure COMP_PPW2, construct an array PP of pointers, such that for each j and l, $0 \leq j < n$ and $1 \leq l < n$, $PP(j, l)$ contains the nil pointer if $l = 1$ or if $P(j, l)$ is false. If $P(j, l)$ is true and $l > 1$, then let $PP(j, l)$ contain a pointer to $PP(j, l \Leftrightarrow 1)$ if $P(j, l \Leftrightarrow 1)$ is true, and to $PP((j \Leftrightarrow 1) \bmod n, l \Leftrightarrow 1)$ otherwise. The computation of PP can be done during the computation of P. Afterwards, if there is an intervalization, then one can be constructed as follows. Start with a j, $0 \leq j < n$ for which $\{v_j, v_{j-1}\} \in E_{\mathrm{E}}$ and $P(j, n \Leftrightarrow 1)$ is true. Then follow the pointers from $PP(j, n \Leftrightarrow 1)$ until the nil pointer is reached, and add edge $\{v_i, v_{i+l}\}$ for each i and l for which $PP(i, l)$ is passed. Note that the nil pointer is reached if the previous pointer pointed to $PP(i, 1)$ for some i such that $\{v_i, v_{i+1}\} \in E_{\mathrm{S}}$.

5 Intervalizing Four-Colored Graphs

For some time, it has been an open problem whether there exist polynomial time algorithms for ICG for some constant number of colors, $k \geq 4$. Older results showed fixed parameter intractability [FHW93, BFH94], but did not resolve the question. Our NP-completeness result resolves the open problem in a negative way (assuming P $\neq$ NP).

Theorem 5.1. *ICG is NP-complete for four-colored graphs.*

Proof. Clearly, ICG $\in$ NP.

To prove NP-hardness, we transform from three-partition, which is strongly NP-complete [GJ79].

> Three-Partition
> **Instance:** Integers $m \in \mathbf{N}$ and $Q \in \mathbf{N}$, a sequence $s_1,\ldots,s_{3m} \in \mathbf{N}$ such that $\sum_{i=1}^{3m} s_i = mQ$, and $\forall_{1\leq i\leq 3m}\ \frac{1}{4}Q < s_i < \frac{1}{2}Q$.
> **Question:** Can the set $\{1,\ldots,3m\}$ be partitioned into m disjoint sets $S_1,\ldots,S_m$ such that $\forall_{1\leq j\leq m} \sum_{i\in S_j} s_i = Q$?

Suppose input $m, Q, s_1, s_2,\ldots, s_{3m} \in \mathbf{N}$ is given. Now, we define a graph $G = (V, E)$, which consists of the following parts (see Figure 4).

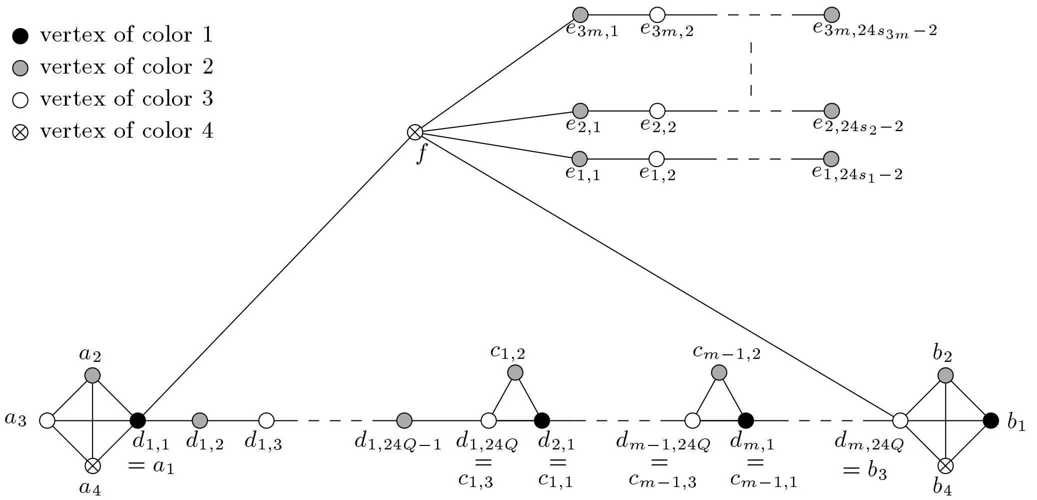

Figure 4: The constructed graph $G = (V, E)$.

Start clique. Take vertices $A = \{a_1, a_2, a_3, a_4\}$. Color vertex a_i with color i ($i = 1, 2, 3, 4$). Add edges between every two vertices in A.

End clique. Take vertices $B = \{b_1, b_2, b_3, b_4\}$. Color vertex b_i with color i ($i = 1, 2, 3, 4$). Add edges between every two vertices in B.

Middle cliques. Take vertices $C = \{c_{i,j} \mid 1 \leq i \leq m \Leftrightarrow 1,\ 1 \leq j \leq 3\}$. Color each vertex $c_{i,j} \in C$ with color j. Make each set $C_i = \{c_{i,1}, c_{i,2}, c_{i,3}\}$ into a clique.

Tracks. Take vertices $D = \{d_{i,j} \mid 1 \leq i \leq m,\ 1 \leq j \leq 24Q\}$. Color each vertex $d_{i,j} \in D$ with color 1 if $j \bmod 3 = 1$, with 2 if $j \bmod 3 = 2$ and with 3 if $j \bmod 3 = 0$. Identify vertex a_1 with $d_{1,1}$, vertex b_3 with $d_{m,24Q}$, and, for all i, $1 \leq i \leq m \Leftrightarrow 1$, identify $d_{i,24Q}$ with $c_{i,3}$, and $d_{i+1,1}$ with $c_{i,1}$. These track vertices form m paths: take edges $\{d_{i,j}, d_{i,j+1}\}$ for all i, j, $1 \leq i \leq m$, $1 \leq j \leq 24Q \Leftrightarrow 1$.

Number representing paths. Take vertices $E = \{e_{l,j} \mid 1 \le l \le 3m,\ 1 \le j \le 24s_l \Leftrightarrow 2\}$. Color each vertex $e_{l,j} \in E$ with color 2 if $j \bmod 3 = 1$, with color 3 if $j \bmod 3 = 2$, and with color 1 if $j \bmod 3 = 0$. For each l, the vertices $E_l = \{e_{l,j} \mid 1 \le j \le 24s_l \Leftrightarrow 2\}$ form a path: add edges $\{e_{l,j}, e_{l,j+1}\}$ for all l, j, $1 \le l \le 3m$, $1 \le j \le 24s_l \Leftrightarrow 3$.

Attachment vertex. Take one vertex f. Color f with color 4. Take edges $\{f, a_1\}$ $\{f, b_3\}$, and for all l, $1 \le l \le 3m$, edge $\{f, e_{l,1}\}$.

The four-colored graph, resulting from this construction, is the graph $G = (V, E)$. Note that the transformation can be done in polynomial time in Q and m.

Claim 5.1. There exists a partition of the set $\{1,...,3m\}$ into sets $S_1,...,S_m$ such that $\sum_{i \in S_j} s_i = Q$ for each j if and only if there is an intervalization of G.

Proof. Suppose that G is a subgraph of a properly colored interval graph. So, we have a proper path decomposition $(V_1,...,V_r)$ of G. We may assume that there are no V_i, V_{i+1} with $V_i \subseteq V_{i+1}$ or $V_{i+1} \subseteq V_i$. (Otherwise, we may omit the smaller of these two sets from the path decomposition and still have a path decomposition of G.)

Note that, by the clique containment lemma (Lemma 2.4), there exist i_0 with $V_{i_0} = A$, and i_1 with $V_{i_1} = B$. Without loss of generality suppose $i_0 < i_1$. If $i_0 \ne 1$, then there exists a $v \in V_{i_0-1}$ with $v \notin A$. Note that such a vertex v has a path to a vertex in B that avoids A. It follows that V_{i_0} must contain a vertex from this path, but this will yield a color conflict with a vertex in A, contradiction. So, $i_0 = 1$. A similar argument shows that $i_1 = r$.

Also, from the clique containment lemma it follows that for each i, $1 \le i \le m \Leftrightarrow 1$, there is a j_i, $2 \le j_i \le r \Leftrightarrow 1$ with $C_i \subseteq V_{j_i}$. We must have $j_1 < j_2 < j_3 < \cdots < j_{m-1}$, otherwise a color conflict will arise between a track vertex and a vertex in a set C_i. Write $j_0 = 1$, $j_m = r$. As there is a path from $d_{1,1}$ to $d_{m,24Q}$ in G that does not contain vertices with color 4 or vertices in E, it follows that each set V_i contains at least one vertex in $C \cup D$ with color 1, 2 or 3.

For each i, $1 \le i \le m$, call the interval $[j_{i-1} + 1, j_i \Leftrightarrow 1]$ the *ith valley*. Each vertex $d_{i,j}$ must be in one or more successive nodes V_α with α in the ith valley. It can not be in another valley, since that gives a color conflict. Note that there are exactly $8Q$ vertices $d_{i,j}$ (for fixed i) with color 2. For a two-colored vertex $d_{i,j}$, we call the interval $\{\alpha \mid d_{i,j} \in V_\alpha\}$ a 2-range. Note that all 2-ranges are disjoint, otherwise we have a color conflict. So, in each valley, we have exactly $8Q$ 2-ranges.

For each l, $1 \le l \le 3m$, look at the vertices E_l. Note that all vertices in E_l must be contained in nodes V_α with all α's in the same valley. Otherwise, the path induced by E_l will cross a middle clique, and we have a color conflict between a vertex in E_l and a vertex in C. Write $S_i = \{l \mid$ vertices in E_l are in sets V_α with α in the ith valley$\}$. We show that $S_1,...,S_m$ is a partition of $\{1,...,3m\}$ such that for each j, $\sum_{i \in S_j} s_i = Q$.

For each edge $\{e_{l,j}, e_{l,j+1}\}$ with $e_{l,j}$ of color 3 (and hence, $e_{l,j+1}$ has color 1), there must be a node α with $\{e_{l,j}, e_{l,j+1}\} \subseteq V_\alpha$. α must be in a 2-range, as otherwise V_α contains a one-colored or three-colored vertex from $C \cup D$, and we have a color conflict.

If there exists an α with $\{e_{l,j}, e_{l,j+1}, d_{i,j'}\} \subseteq V_\alpha$, with $d_{i,j'}$ of color 2, then we say that the 2-range of $d_{i,j'}$ contains the 1-3-E-edge $\{e_{l,j}, e_{l,j+1}\}$.

Claim 5.2. No 2-range contains two or more 1-3-E-edges.

Proof. Suppose $\{e_{l_1,j_1}, e_{l_1,j_1+1}\}$ and $\{e_{l_2,j_2}, e_{l_2,j_2+1}\}$ are distinct 1-3-E-edges, and there is a $d_{i,j'}$ such that $\{e_{l_1,j_1}, e_{l_1,j_1+1}, d_{i,j'}\} \subseteq V_\alpha$, $\{e_{l_2,j_2}, e_{l_2,j_2+1}, d_{i,j'}\} \subseteq V_\beta$. Suppose w.l.o.g. that $\alpha < \beta$. Note that both $v = e_{l_1,j_1}$ and $w = e_{l_1,j_1+1}$ are adjacent to a two-colored vertex. Let $[\gamma, \delta]$ be the 2-range of $d_{i,j'}$. Note that $\gamma \leq \alpha < \beta \leq \delta$. If $V_{\gamma-1}$ contains a one-colored vertex from $C \cup D$, then consider the 1-colored vertex w. It can not belong to $V_{\gamma-1}$ and it cannot belong to V_β. So, if $w \in V_\epsilon$, then $\gamma \leq \epsilon \leq \delta$. Hence, there cannot be a set V_ϵ that contains w and its two-colored neighbor e_{l_1,j_1+2}, contradiction. If $V_{\gamma-1}$ does not contain a one-colored vertex from $C \cup D$, then it contains a three-colored vertex from $C \cup D$, and by considering v and using a similar argument, also a contradiction arises. □

Let $1 \leq i \leq m$. Suppose $S_i = \{l_1, l_2, ..., l_t\}$. Note that $E_{l_1} \cup \cdots \cup E_{l_t}$ induces $8s_{l_1} \Leftrightarrow 1 + 8s_{l_2} \Leftrightarrow 1 + \cdots + 8s_{l_t} \Leftrightarrow 1$ 1-3-E-edges. As there are $8Q$ 2-ranges in a valley, we must have

$$8(s_{l_1} + s_{l_2} + \cdots s_{l_t}) \Leftrightarrow t \leq 8Q$$

By noting that each $s_l \geq Q/4 + 1/4$, it follows that $8(Q/4 + 1/4)t \Leftrightarrow t \leq 8Q$, so $t \leq 3$, and that hence also, by integrality,

$$8(s_{l_1} + s_{l_2} + \cdots s_{l_t}) \leq 8Q$$

So, we have a partition of $\{1, ..., 3m\}$ into sets $S_1, ..., S_m$, such that for all $j, 1 \leq j \leq m$, $\sum_{i \in S_j} s_i \leq Q$. As $\sum_{j=1}^{m} \sum_{i \in S_j} s_i = mQ$, it follows that for all j, $1 \leq j \leq m$, $\sum_{i \in S_j} s_i = Q$.

Now, suppose $S_1, S_2, ..., S_m$ is a partition of $\{1, ..., 3m\}$, such that for all j, $1 \leq j \leq m$, $\sum_{i \in S_j} s_i = Q$. We will give a path decomposition $(V_1, ..., V_r)$ of $G = (V, E)$, such that no V_i contains two vertices of the same color. We leave most of the easy verification that the given path decomposition fulfills the requirements to the reader.

Take $t = 48Q$, $r = mt + 1$.

Take $V_1 = A$, $V_r = B$.

For each vertex $c_{i,j} \in C$, put $c_{i,j}$ in set V_{ti+1}.

For each vertex $d_{i,j} \in D$, put $d_{i,j}$ in sets $V_{t(i-1)+2j-1}$, $V_{t(i-1)+2j}$, and $V_{t(i-1)+2j+1}$. (Identified vertices are just put in every set, indicated by their 'different names'; one easily observes that these are consecutive sets.)

For each i, $1 \leq i \leq m$, suppose $S_i = \{l_1, l_2, l_3\}$. Put vertex $e_{l_1,1}$ in set $V_{t(i-1)+2}$. For all j, $2 \leq j \leq 24s_{l_1} \Leftrightarrow 2$, put vertex $e_{l_1,j}$ in sets $V_{t(i-1)+2j-2}$, $V_{t(i-1)+2j-1}$, $V_{t(i-1)+2j}$. For all j, $1 \leq j \leq 24s_{l_2} \Leftrightarrow 2$, put vertex $e_{l_2,j}$ in sets $V_{t(i-1)+48s_{l_1}+2j-2}$, $V_{t(i-1)+48s_{l_1}+2j-1}$, $V_{t(i-1)+48s_{l_1}+2j}$. For all j, $1 \leq j \leq 24s_{l_3} \Leftrightarrow 2$, put vertex $e_{l_3,j}$ in sets $V_{t(i-1)+48s_{l_1}+48s_{l_2}+2j-2}$, $V_{t(i-1)+48s_{l_1}+48s_{l_2}+2j-1}$, $V_{t(i-1)+48s_{l_1}+48s_{l_2}+2j}$.

Finally, put f in all sets $V_2, ..., V_{r-1}$.

A straightforward, but somewhat tedious verification shows that the resulting path decomposition is indeed a path decomposition of G, and that no set V_i contains two different vertices with the same color.

□

As three-partition is strongly NP-complete and our transformation is polynomial in Q and m, the claimed theorem now follows. □

Note that we even proved a slightly stronger result.

Corollary 5.1. *ICG is NP-complete for four-colored graphs G, with the property that there is one color that is only given to three vertices of G.*

Note furthermore that the constructed graph is connected, hence ICG is NP-complete for connected four-colored graphs. With this result, we can prove the following theorem.

Theorem 5.2. *ICG is NP-complete for biconnected five-colored graphs.*

Proof. Clearly, ICG for biconnected five-colored graphs is in NP.

To prove NP-hardness, we transform from ICG for connected four-colored graphs. Let $G = (V, E)$ be a connected graph, $c : V \to \{1, 2, 3, 4\}$ a four-coloring. Then we construct a graph $G' = (V', E')$ with five-coloring $c' : V' \to \{1, 2, 3, 4, 5\}$ as follows. Let $V' = V \cup \{x\}$, where x is a new vertex which is not in V, and let $E' = E \cup \{ \{v, x\} \mid v \in V \}$. Furthermore, for all $v \in V$, let $c'(v) = c(v)$, and let $c'(x) = 5$. Note that G' is biconnected, and that the transformation can be done in polynomial time.

It is easy to show that there is a proper path decomposition of G if and only if there is a proper path decomposition of G'. □

6 Conclusions and Remarks

In this paper, we have given an $O(n^2)$ time algorithm to determine whether we can add edges to a given biconnected three-colored graph such that it becomes a properly colored interval graph. The algorithm can be modified such that it outputs an intervalization, if existing, and still uses quadratic time.

To get a faster algorithm for the problem considered in this paper might well be a hard problem. It seems that even the simplest cases, e.g., when G is a simple cycle, need $O(n^2)$ time to resolve, and might well already capture the main difficulties for speed-up.

We have used the algorithm for biconnected three-colored graphs to obtain an $O(n^2)$ algorithm for general three-colored graphs ([BdF95]). This algorithm consists of an extensive case analysis. In each of the cases, a modified version of our algorithm for biconnected graphs is used. It seems that if a faster algorithm for ICG on biconnected three-colored graphs is found, then this algorithm can be used to construct an equally fast algorithm for general three-colored graphs.

We have shown that ICG is NP-complete for four or more colors. We feel however that the graphs, arising in the reduction of this proof, will not be typical for the type of colored graphs, arising in the sequence reconstruction application. It may well be that special cases of ICG, which capture characteristics of the application data, have efficient algorithms. Further research could perhaps give new meaningful results here.

The problem INTERVALIZING SANDWICH GRAPHS is a generalization of ICG. We surmise that our algorithm for ICG can be modified such that it solves the problem of intervalizing sandwich graphs with clique size at most three in $O(n^2)$ time, where n is the number of vertices of the sandwich graph.

A special case of UNIT-INTERVALIZING SANDWICH GRAPHS (UISG) is the problem UNIT-INTERVALIZING COLORED GRAPH (USCG), which asks whether there exists a supergraph G' of a given graph G, such that G' is a unit interval graph, and is properly colored by a given coloring c for G. The $O(n^{k-1})$ algorithm of [KS93, KST94] for UISG with maximum clique size k can also be used for UICG with k colors. For $k = 3$, this gives an $O(n^2)$ time algorithm. We expect that our algorithm for ICG can be used to obtain a linear time algorithm for this problem with $k = 3$.

References

[BdF95] H. L. Bodlaender and B. de Fluiter. Intervalizing k-colored graphs. Technical Report UU-CS-1995-15, Department of Computer Science, Utrecht University, Utrecht, 1995.

[BFH94] H. L. Bodlaender, M. R. Fellows, and M. T. Hallett. Beyond *NP*-completeness for problems of bounded width: Hardness for the W hierarchy. In *Proceedings of the 26th Annual Symposium on Theory of Computing*, pages 449–458, New York, 1994. ACM Press.

[BFW92] H. L. Bodlaender, M. R. Fellows, and T. J. Warnow. Two strikes against perfect phylogeny. In *Proceedings 19th International Colloquium on Automata, Languages and Programming*, pages 273–283, Berlin, 1992. Springer Verlag, Lecture Notes in Computer Science, vol. 623.

[BK93] H. L. Bodlaender and T. Kloks. A simple linear time algorithm for triangulating three-colored graphs. *J. Algorithms*, 15:160–172, 1993.

[BM93] H. L. Bodlaender and R. H. Möhring. The pathwidth and treewidth of cographs. *SIAM J. Disc. Meth.*, 6:181–188, 1993.

[FHW93] M. R. Fellows, M. T. Halett, and H. T. Wareham. DNA physical mapping: Three ways difficult (extended abstract). In Thomas Lengauer, editor, *Proceedings 1st Annual European Symposium on Algorithms ESA'93*, pages 157–168. Springer Verlag, Lecture Notes in Computer Science, vol. 726, 1993.

[GJ79] M. R. Garey and D. S. Johnson. *Computers and Intractability, A Guide to the Theory of NP-Completeness.* W.H. Freeman and Company, New York, 1979.

[GKS94] M. C. Golumbic, H. Kaplan, and R. Shamir. On the complexity of DNA physical mapping. *Advances in Applied Mathematics*, 15:251–261, 1994.

[IS93] R. Idury and A. Schaffer. Triangulating three-colored graphs in linear time and linear space. *SIAM J. Disc. Meth.*, 2:289–293, 1993.

[KS93] H. Kaplan and R. Shamir. Pathwidth, bandwidth and completion problems to proper interval graphs with small cliques. Technical Report 285/93, Inst. for Computer Science, Tel Aviv University, Tel Aviv, Israel, 1993. To appear in SIAM J. Comput.

[KST94] H. Kaplan, R. Shamir, and R. E. Tarjan. Tractability of parameterized completion problems on chordal and interval graphs: Minimum fill-in and physical mapping. In *Proceedings of the 35th annual symposium on Foundations of Computer Science (FOCS)*, pages 780–791. IEEE Computer Science Press, 1994.

[KW92] S. Kannan and T. J. Warnow. Triangulating 3-colored graphs. *SIAM J. Disc. Meth.*, 5:249–258, 1992.

[MWW94] F. R. McMorris, T. J. Warnow, and T. Wimer. Triangulating vertex-colored graphs. *SIAM J. Disc. Meth.*, 7(2):296–306, 1994.

[NON94] S.-I. Nakano, T. Oguma, and T. Nishizeki. A linear time algorithm for c-triangulating three-colored graphs. *Trans. Institute of Electronics, Information and Communication, Eng., A.*, 377-A(3):543–546, 1994. In Japanese.