

Precoloring extension. I. Interval graphs

M. Biró

*Computer and Automation Institute, Hungarian Academy of Sciences, Budapest, Kende u
13–17, H-1111 Hungary*

M. Hujter

Institute of Mathematics, University of Miskolc, Miskolc-Egyetemváros, H-3515 Hungary

Zs. Tuza

*Computer and Automation Institute, Hungarian Academy of Sciences, Budapest, Kende u
13–17, H-1111 Hungary*

Received 8 March 1991

Revised 31 October 1991

Abstract

Biró, M., M. Hujter and Zs. Tuza, Precoloring extension. I. Interval graphs, *Discrete Mathematics* 100 (1992) 267–279.

This paper is the first article in a series devoted to the study of the following general problem on vertex colorings of graphs. Suppose that some vertices of a graph G are assigned to some colors. Can this ‘precoloring’ be extended to a proper coloring of G with at most k colors (for some given k)? This question was motivated by practical problems in scheduling and VLSI theory. Here we investigate its complexity status for interval graphs and for graphs with a bounded treewidth.

1. Introduction

We consider finite undirected graphs $G = (V, E)$ with vertex set V and edge set E . The *clique number* or *maximum clique size* and the *chromatic number* of G are denoted by $\omega(G)$ and $\chi(G)$, respectively. For any vertex subset $W \subseteq V$, G_W denotes the subgraph induced by W . By definition, for a given integer $k \geq 2$, a (proper) k -coloring is a function $f: V \rightarrow \{1, 2, \dots, k\}$ such that $uv \in E$ implies $f(u) \neq f(v)$.

The problem we raise and investigate in this paper is called the PRECOLORING EXTENSION problem, or PrExt in short. PrExt is more general than the usual CHROMATIC NUMBER problem and less general than

LIST-COLORING [21]. (The latter has been studied extensively for *line graphs* [5, 9, 15, 13].) PrExt can be formulated as follows.

Instance. An integer $k \geq 2$, a graph $G = (V, E)$ with $|V| \geq k$, a vertex subset $W \subseteq V$, and a proper k -coloring of G_W .

Question. Can this k -coloring be extended to a proper k -coloring of the whole graph G ?

Because of their theoretical and practical importance, in this introductory paper we focus on interval graphs. (A graph $G = (V, E)$ is an *interval graph* if its vertices can be represented by open real intervals in such a way that two vertices are adjacent if and only if the corresponding intervals intersect each other.) It is well known that the chromatic number of an interval graph can be determined by $O(|V| + |E|)$ time. It will turn out that PrExt is much harder.

In order to indicate the difficulties, first let us consider an example. Let $k \geq 2$ be any fixed integer, and let $n \geq 2k$ be an arbitrary multiple of k . We define a graph $G = (V, E)$ on the vertex set $V = \{1, 2, \dots, n\}$ as follows: let $uv \in E$ if and only if $u, v \in \{1, 2, \dots, n\}$ and $0 < |u - v| < k$. Observe that G is an interval graph with $\chi(G) = \omega(G) = k$. (An interval representation can be given as follows: For any $i \in \{1, 2, \dots, n\}$, let $a_i = i$ and $b_i = i + k$. Now $ij \in E$ if and only if $i, j \in \{1, 2, \dots, n\}$ and $(a_i, b_i) \cap (a_j, b_j) \neq \emptyset$.) On the other hand, assigning color 1 to vertices 1 and n , we obtain a precoloring which cannot be extended to a proper k -coloring of G . This example also shows that the answer to PrExt may depend on each edge of the graph since by deleting any edge, the precoloring becomes extendible.

Our paper is organized as follows. At the end of the Introduction we give the basic notions related to PrExt. The applications of PrExt in scheduling and VLSI theory are described in Sections 2 and 3, respectively. In Section 4, we prove that PrExt is polynomially solvable on interval graphs when every color is used at most once in the precoloring. On the other hand, the problem becomes NP-complete when we allow the colors to be used twice, as shown in Section 5. Finally, if the number k of colors is fixed, then PrExt is polynomially solvable. This fact is proved in a much more general form, for graphs of bounded treewidth, in Section 6.

We note that PrExt is closely related to many interesting concepts of combinatorics, including partial Latin squares, integer-valued multicommodity flows, bipartite matching, perfect graphs, etc. These connections will be discussed in our forthcoming papers [10–12].

Basic notions. For an instance of PrExt we say that the number k is the *color bound*, and G is a *precolored* or a *partially k -colored graph*. The vertices of W and $V - W$ are called *precolored* and *precolorless*, respectively. If G is completely colored, i.e., $W = V$, the answer to PrExt is ‘yes’ needing no proof. More

generally, the same is true if the number of different colors assigned to the precolored vertices is at most $k - |V - W|$. Given a precoloring φ on W , the *precolored classes* are the sets

$$C_i = \{x \in W : \varphi(x) = i\}, \quad i = 1, 2, \dots, k$$

Given a nonnegative integer d , the subproblem d -PrExt is defined as the problem in which the instances of PrExt are restricted to those partially k -colored graphs where the size of each precolored class is at most d . Note that 0-PrExt is the usual *chromatic number* problem, i.e., 'Is $\chi(G) \leq k$?'.

2. PrExt and scheduling

The following scheduling problem arose in 'real life' at MALÉV, the Hungarian Airlines. Suppose we have n jobs, $J_1, J_2, \dots, J_n$. For each job J_i , an open interval (a_i, b_i) is fixed where $0 \leq a_i < b_i$ are integers. Job J_i must be processed in this time interval. In addition, we have k identical machines, $M_1, M_2, \dots, M_k$. (Here $k \geq 2$ is a fixed integer.) The jobs are partitioned into two classes: For $i = 1, 2, \dots, m$, some machine M_j is already assigned to job J_i , and for $i = m + 1, m + 2, \dots, n$, we have to assign some machine M_j to job J_i . Any machine can process at most one job at any time. Here an appropriate assignment means a feasible scheduling. From the many possibilities (their number is bounded above by k^{n-m}), we have to choose an appropriate one, or we have to prove that it does not exist.

In the case of this practical problem of MALÉV, the machines are airplanes of a given type. The typical value of k is five, and the time horizon is one week; the integers a_i and b_i mean specific hours of the week. The jobs $J_1, J_2, \dots, J_m$ correspond to the time intervals of the required *maintenances* of the planes. Typically, each plane requires maintenance once or twice a week; hence m is usually between 5 and 10. It is important to note that for each maintenance time interval, the specific airplane to be checked and/or repaired in that time interval is fixed. The maintenance process is done according to some prescribed plan.

The jobs $J_{m+1}, J_{m+2}, \dots, J_n$ correspond to the time intervals of *flights* determined by the seasonal schedule being valid on the week in question. These time intervals include not only the hours when the plane is in the air but also the hours the plane has to spend at the airports. Each flight starts at Budapest Airport, and this airport is the terminal, too. There is only one degree of freedom: each flight can be performed by any of the planes $M_1, M_2, \dots, M_k$. Certainly, in each hour each plane can perform at most one flight. Typically, the number of flights is around 50 a week for each type of airplanes.

This scheduling problem is actually PrExt for interval graphs. The vertices are the jobs $J_1, J_2, \dots, J_n$. To color a vertex means the assignment of some plane to the corresponding time interval. The precolored vertices are those which

correspond to the maintenance intervals, and the precolorless vertices are those which correspond to the flights. Two vertices are non-adjacent if and only if the corresponding intervals are disjoint. (This means that the only restriction for assigning the same plane to two distinct flights, or to a flight and to a maintenance time interval, is that the corresponding intervals must not overlap.)

3. PrExt and VLSI theory

The strong connections between graph theory and the theory of Very-Large-Scale-Integration (VLSI) technology have been known from the very beginning of the latter. We refer the reader to [16].

Using the terminology and notation of [16], we present a model of a special layout problem in VLSI theory. This is called '*multiterminal net channel routing in the knock-knee layout mode*'.

The *uniform grid* of the plane is the set of lines $\{x = i: i \text{ is integer}\}$ and $\{y = j: j \text{ is integer}\}$. A *k-track channel* is a horizontal strip of the uniform grid delimited by the lines $y = 0$ and $y = k + 1$. These two delimiter lines are called *shores*. The k lines of the form $\{y = j\}$, $j = 1, 2, \dots, k$, are the *tracks* of the channel. The number k is called the *capacity* of the channel. A grid point on either shore is called a *terminal*. The channel is considered as an infinite graph with grid points as vertices and the segments of length one as edges.

A (multiterminal) *net* is a pair of (not simultaneously empty) sequences of terminals on the upper and lower shores, respectively. A net is denoted by $N = ((s_1, s_2, \dots, s_p), (t_1, t_2, \dots, t_q))$ with $s_1 < s_2 < \dots < s_p$, $t_1 < t_2 < \dots < t_q$, $p + q > 0$. Such a net is represented by a horizontal *spoked segment* from $\min(s_1, t_1)$ to $\max(s_p, t_q)$, with a down-directed spoke for each s_i and an up-directed spoke for each t_j . An example can be seen in Fig. 1.

An instance of the channel routing problem is a collection of nets no two of which share a terminal. A *layout* is a one-to-one assignment of pairwise edge-disjoint connected subgraphs $w_1, w_2, \dots, w_n$ of the channel to the nets such that each w_i has a vertex at each terminal of the corresponding net.

In the discussion below we restrict our considerations to such layouts where no w_i has edges in more than one track; therefore each w_i can be viewed as a spoked segment put on one of the tracks, and the vertical spokes connect the horizontal segments with the terminals. In this sense each layout is actually a proper k -coloring of the interval graph determined by the intervals corresponding

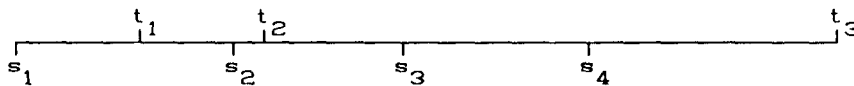


Fig. 1.

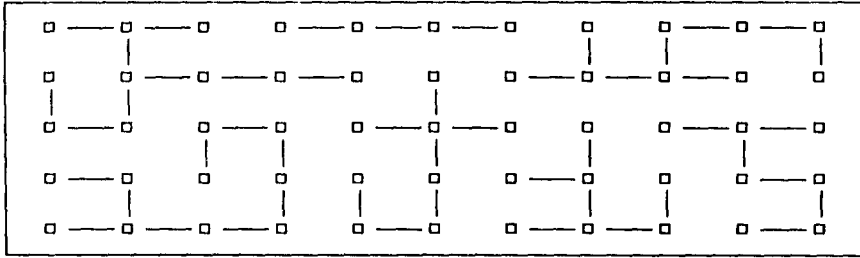


Fig. 2.

to the horizontal parts of the spoked segments. In the case of many practical problems, for some nets the container tracks are prescribed. This leads to PrExt on interval graphs.

In another special case studied by Bisztray and Frank [4], no track is prescribed, but only one specified segment of each track is allowed to be used in the layout. Let $[c_j, d_j]$ denote the allowed segment corresponding to the j th track where $0 < c_j < d_j$ are given integers, $j = 1, 2, \dots, k$. Now we adjoin $2k$ further intervals to the interval representation of the interval graph whose vertices are the allowed segments. Let $D = \max\{d_j: j = 1, 2, \dots, k\}$ and define the additional intervals as $[0, c_j)$ and $(d_j, D + 1]$, $j = 1, 2, \dots, k$. If the j th intervals in both sequences are precolored with color j , then we gain a partially k -colored graph where any extension corresponds to some feasible layout and vice versa. Therefore the layout problem at hand is a particular case of 2-PrExt on interval graphs. We will study this problem in Section 5.

At the end of this section we show an unexpected application of PrExt to circuit testing. Its idea is due to [6] (see also [14]). When a circuit board is constructed, connections are made on the reverse side of the circuit board joining certain nodes into nets which are electronically common. Fig. 2 shows the picture of the reverse side of such a circuit board.

If the solder was carelessly applied, it is possible that some extraneous connections were made. Hence we should test every pair of nets for such shorts; however, there is a faster method. Let V denote the set of nets, and define a graph $G = (V, E)$ where E consists of those pairs of nets which are, in some sense, close to each other. (Those pairs of nets which are some distance apart on the board need not be tested because of the physical impossibility of such extraneous connections.) The crucial idea is this: if G is properly k -colored (where k is a small integer, e.g. 5), then constructing for each color class a yoke of wires which contact simultaneously all nets in that color class, only $\binom{k}{2}$ tests will reveal any short present. However, if some subset of the net set must be contacted with the same yoke of wires (because of some physical reasons), we obtain an instance of PrExt to solve before designing the tests.

4. 1-PrExt for interval graphs

The objective of this section is to prove the following result.

Theorem 4.1. *1-PrExt is polynomially solvable on interval graphs.*

Proof. We use the following equivalent definition of interval graphs: A (finite) graph $G = (V, E)$, $V = \{1, 2, \dots, n\}$, is an interval graph if there are nonnegative integers $a_i < b_i$, $i = 1, 2, \dots, n$, such that $ij \in E$ holds if and only if the open intervals (a_i, b_i) and (a_j, b_j) intersect. Without loss of generality, for an appropriate order of vertices we may assume that $0 \leq a_i < b_i = i$, $i = 1, 2, \dots, n$. Such an order and such numbers a_i and b_i can be found in polynomial time from the original representation of the interval graph.

Let $G = (\{1, 2, \dots, n\}, E)$ be an interval graph given by the sequence $a_1 = 0, a_2, \dots, a_n$ such that $a_i < i$, $i = 1, 2, \dots, n$ and suppose that $W \subseteq \{1, 2, \dots, n\}$ is the set of its precolored vertices (distinct vertices are assigned to distinct colors). Denote by k the color bound of 1-PrExt on G . The case $k < \omega(G)$ is easy because in this case $\omega(G)$ can be computed in linear time and the answer for 1-PrExt is trivially ‘no’. Therefore, without loss of generality we can assume that $k \geq \omega(G)$.

We construct an acyclic directed graph $D = D_{G,W,k}$ on the slightly larger vertex set $\{0, 1, \dots, n, n+1\}$ as follows. There will be three types of arcs: *original*, *auxiliary*, and *modified*. The original arcs will be of the form $\overrightarrow{a_i i}$, $i \in \{1, 2, \dots, n\} - W$. Hence, these arcs are in one-to-one correspondence with the intervals representing the precolorless vertices of G . The auxiliary arcs will be of the form $\overrightarrow{(i-1)i}$, $i = 1, 2, \dots, n+1$. Finally, the modified arcs are of the form $\overrightarrow{a_i(n+1)}$, $i \in W$.

Now we assign a nonnegative integer capacity to each arc. The original and the modified arcs all get capacity equal to one. The capacity of an auxiliary arc $\overrightarrow{(i-1)i}$, $i = 1, 2, \dots, n$, will be defined as $k - |\{a_j : a_j < i \leq j\}|$, and the capacity of $\overrightarrow{n(n+1)}$ will be $k - |W|$.

We show that there exists a flow of value k from 0 to $n+1$ in D if and only if the precoloring can be extended to a proper k -coloring f of G . Suppose first that the precoloring φ can be extended to some $f : V \rightarrow \{1, 2, \dots, k\}$. If color i does not occur in the precoloring, then we set $V_i = f^{-1}(i)$. Otherwise, if say the vertex corresponding to (a_j, j) has color i in W , then we define V_i as the set of those vertices in $f^{-1}(i)$ which correspond to the intervals whose right endpoints are at most j . It is easy to see that the union of the set of arcs corresponding to the vertices of the V_i and the set of the auxiliary arcs admits a flow of value k from 0 to $n+1$ in D .

In order to prove the converse statement, suppose that there is a flow F of value k . Since the capacity of each arc is an integer, we may assume that F has an integer value on each arc. Let D' be the directed graph obtained from D by replacing each auxiliary arc $\overrightarrow{(i-1)i}$ ($i = 1, 2, \dots, n+1$) of capacity c by c distinct

arcs from $i - 1$ to i . The existence of F in D implies the existence of k arc-disjoint paths $P_1, P_2, \dots, P_k$ from 0 to $n + 1$ in D' . Obviously, each P_i contains at most one modified arc. Without loss of generality, we may assume that the color of a vertex $w \in W$ is i if and only if the modified arc corresponding to w belongs to P_i . (Every modified arc belongs to some P_i because the arc $\overrightarrow{n(n+1)}$ has capacity $k - |W|$.)

Let us return to the interval representation of G , adding as many unit intervals $(i - 1, i)$ to it as the capacity of $\overrightarrow{(i-1)i}$ is in D , $i = 1, 2, \dots, n$. In this interval structure S , the number of intervals containing any non-integer point between 0 and n is equal to k . Every arc of D' , other than $\overrightarrow{n(n+1)}$, corresponds to an interval of S . Therefore each P_i corresponds to a sequence of consecutive intervals in S , the first one starting at 0; denote their convex hull by H_i . We delete the intervals of every P_i from S , and adjoin the H_i as new intervals. The interval structure obtained will be denoted by S' .

The intersection graph G' of S' has $\omega(G') = k$ and therefore $\chi(G') = k$; moreover, a proper k -coloring f' of G' can be found in polynomial time. Since the H_i are intersecting, they get distinct colors. We assume that $f'(H_i) = i$. For $v \in V$, let $f(v) = i$ if and only if either P contains the (original or modified) arc corresponding to v , or the interval representing v in S' is disjoint from H_i and gets color i in f' . Observe that f is a proper k -coloring with the required properties.

Since the flow F can be found (or its non-existence can be proved) in polynomial time, the theorem follows. $\square$

5. 2-PrExt for interval graphs

We have already seen some instances of 2-PrExt for interval graphs (namely, the graph given in the Introduction and the instances of the problem of [4] as defined in Section 3).

The main objective of this section is to prove the following result.

Theorem 5.1. *2-PrExt on interval graphs is NP-complete.*

Proof. We construct instances of 2-PrExt from an arbitrary instance of the chromatic number problem on *circular-arc* graphs. By definition, circular-arc graphs are the intersection graphs of closed arcs of a circle (cf. [8]). Note that each interval graph is a circular-arc graph (but not conversely, e.g., every chordless cycle is a circular-arc graph). In the present context the most important fact is that for unbounded k , the question ‘Is $\chi(G) \leq k$?’ is NP-complete on circular-arc graphs [7].

Let us consider an arbitrary circular-arc graph $G = (V, E)$ (together with its arc-representation) and arbitrary integer k . Without loss of generality we can

assume that $V = \{1, 2, \dots, n\}$, and the arcs are some paths of the cycle-graph $C = (\{1, 2, \dots, m\}, \{12, 23, 34, \dots, (m-1)m, m1\})$ where m is a sufficiently large integer. We can also assume that these arcs all contain at least 2 and at most $m-1$ vertices of C . Then the set of those arcs which contain the edge $m1$ of C corresponds to a clique in G . We can assume that the vertices of this clique are $1, 2, \dots, t$ (for some $t \leq \omega(G)$). Recall that each $i \leq t$ is represented by a path P_i in C .

We construct a new graph G' from G as follows. Split each P_i ($1 \leq i \leq t$) into two shorter paths P'_i and P''_i by deleting its edge $m1$. In this way we obtain an interval graph represented by $n+t$ intervals as subpaths of $C - \{m1\}$. Now we precolor each pair of vertices corresponding to P'_i and P''_i , $i = 1, 2, \dots, t$, with color i . The other vertices of G' remain precolorless. Therefore we obtain an instance of 2-PrExt with color bound k . Observe that this precoloring of G' can be extended to a proper k -coloring if and only if $\chi(G) \leq k$.

It was proved by Tucker [20] that circular-arc graphs can be recognized in polynomial time, and an arc-representation can also be found in polynomial time. Moreover, it is obvious that G' can be obtained from G in linear time. Thus the NP-completeness of 2-PrExt follows. $\square$

We have to emphasize that the color bound may vary in the above theorem. If the color bound is fixed (or equivalently, bounded), we can prove the following result.

Theorem 5.2. *If the color bound is fixed, PrExt on circular-arc graphs is polynomially solvable.*

This theorem follows from the results of the next section.

6. Graphs with a bounded treewidth

The concept of treewidth is introduced in [17] through ‘tree decompositions’ of graphs. (Some alternative definitions can be found in [1].) Given a graph $G = (V, E)$, a *tree decomposition* of G is a pair

$$(\{X_i: i \in I\}, T = (I, F))$$

where T is a tree (with vertex set I and edge set F), each X_i is a subset of V , $\bigcup_{i \in I} X_i = V$, each edge of G is contained in some X_i , and if $p, q, r \in I$ and q lies on the path from p to r in T then $X_p \cap X_r \subseteq X_q$. The *treewidth* of G , denoted by $w(G)$, is the smallest integer w for which there exists a tree decomposition of G such that $|X_i| \leq w+1$ for all $i \in I$. Note that for every $v \in V$, the subgraph T_v induced by $\{i \in I: v \in X_i\}$ in T is a non-empty subtree of T and, for $u, v \in V$, $T_u \cap T_v$ is a subtree of T as well (non-empty whenever $uv \in E$).

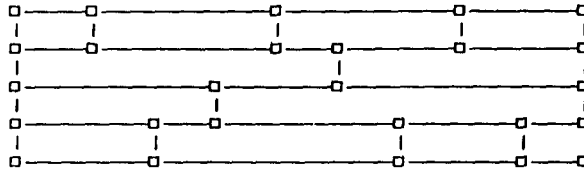


Fig. 3.

In this section our aim is to investigate PrExt for graphs G with $w(G) \leq w$ for some fixed w . First of all, we note that there are lots of well-studied classes of graphs with a bounded treewidth. They include graphs with a bounded *bandwidth*, graphs with a bounded *cutwidth*, *outerplanar graphs*, the so-called *k-outerplanar graphs*, graphs with a bounded *genus* and bounded *disk dimension*, and chordal graphs with a bounded maximum clique size. (A graph is *chordal* if it contains no induced cycle of length greater than three.)

It is a hard problem to characterize graphs with a given treewidth. It has been done, however, for $w(G)$ small. The class of graphs G with $w(G) \leq 1$ and $w(G) \leq 2$ coincides with the class of forests and of series-parallel graphs, respectively [19]. Further interesting examples for $w(G) = 3$ are the Halin graphs. (A plane graph G is called a *Halin graph* if its dual has a dominating vertex; or equivalently, G is the union of a forest F and a cycle C that contains all leaves of F in the appropriate order [3]. A Halin graph is shown in Fig. 3.)

The main result of this section is the following.

Theorem 6.1. *For graphs with bounded treewidth, PrExt is polynomially solvable if the color bound is fixed.*

Without restrictions on the color bound, at the moment we do not know the complexity status of PrExt for graphs with bounded treewidth.

Problem 6.2. Is PrExt polynomially solvable on graphs of bounded treewidth, for an arbitrary color bound?

For trees (i.e., for $w(G) = 1$) this is indeed the case, as shown in [11].

We prove Theorem 6.1 in a slightly more general form, for ‘topological intersection graphs’ of graphs with a bounded treewidth. First let us recall a definition introduced by Scheffler.

Definition [18]. Let H be a fixed graph. A graph $G = (V, E)$ is an *intersection graph* of H if the vertices $v \in V$ can be assigned to connected subgraphs H_v of H in such a way that two vertices $u, v \in V$ are adjacent if and only if H_u and H_v share a vertex.

A *subdivision* H' of a graph H is obtained when the edges of H are replaced by *internally disjoint* paths of arbitrary lengths. (An edge uv of H corresponds to an $u - v$ path in H' .) Note that a graph H is said to be a topological subgraph of a graph H' if a subdivision of H is isomorphic to a subgraph of H' .

We now introduce the following concept.

Definition. A graph $G = (V, E)$ is a *topological intersection graph* of a graph H if G is an intersection graph of some *subdivision* of H .

In this way, interval graphs and circular-arc graphs can be described as the topological intersection graphs of a single edge K_2 and a triangle K_3 , respectively. Moreover, every chordal graph can be represented as the intersection graph of subtrees of a tree. Recalling that the members of those classes can be recognized in polynomial time, we raise the following problem.

Problem 6.3. Let H be an arbitrary fixed graph. Is there a polynomial algorithm testing whether a given graph $G = (V, E)$ is a topological intersection graph of H ?

The following observation is simple but useful.

Proposition 6.4. *Every graph is a topological intersection graph of itself.*

Proof. Let $H(V, E)$ be any graph. Let H' be the subdivision of H in which each edge $e = uv \in E$ is replaced by a path $P_e = uev$ of length two (i.e., the vertex set of H' is $V \cup E$). For $v \in V$, let H_v be the star with center v in H' , with vertex set $\{v\} \cup \{e_{uv} : e_{uv} = uv \in E\}$ and with edge set $\{ve_{uv} : e_{uv} = uv \in E\}$. For isolated vertices $v \in V$, define $H_v = \{v\}$. Then $H_u \cap H_v \neq \emptyset$ if and only if $uv \in E$. $\square$

For treewidth, the following relation holds.

Proposition 6.5. *If H' is a subdivision of H , then $w(H') = w(H)$.*

Proof. Clearly, H' is a forest if and only if so is H . Hence, we can assume $w(H) \geq 2$ and $w(H') \geq 2$. Moreover, applying induction on $|V(H')| - |V(H)|$, it suffices to prove the proposition for the case when H' is obtained from H by an elementary subdivision, i.e., when just one edge $e = uv$ is replaced by a path $P = uev$ of length two.

To prove $w(H') \leq w(H)$, let $(\{X_i : i \in I\}, T = (I, F))$ be a tree decomposition of H with $\max |X_i| = w(H) + 1 \geq 3$. Then $u, v \in X_j$ for some $j \in I$. Adjoin a new vertex j' and the edge jj' to T , and define $X_{j'} = \{u, v, e\}$. Then $(\{X_i : i \in I \cup \{j'\}\}, T' = (I \cup \{j'\}, F \cup \{jj'\}))$ is a tree decomposition of H' with $w(H') \leq w(H)$.

To prove the converse inequality $w(H) \leq w(H')$, let $(\{X_i: i \in I\}, T = (I, F))$ be a tree decomposition of H' with $\max |X_i| = w(H') + 1$. Replace e by v in each X_i containing e , to obtain sets X'_i . Then a tree decomposition of H is defined, because:

- (1) there is a $j \in I$ with $u, e \in X_j$,
- (2) $\bigcup X'_i = V(H') - \{e\} = V(H)$, and
- (3) the requirement on $X'_p \cap X'_r \subseteq X'_q$ is satisfied since $e, v \in X_j$, for some $j' \in I$, so that the union of subtrees induced by $\{i \in I: e \in X_i\}$ and $\{i \in I: v \in X_i\}$ is again a subtree of T .

It is also clear that $|X'_i| \leq |X_i|$ for all $i \in I$, thus $w(H) \leq w(H')$. $\square$

By Propositions 6.4 and 6.5, Theorem 6.1 is implied by the following more general result.

Theorem 6.6. *Let k and w be fixed natural numbers. Then PrExt with color bound k is polynomially solvable on the topological intersection graphs of graphs H with treewidth $w(H) \leq w$.*

Proof. Let $G = (V, E)$ be a topological intersection graph of H . We put $n = |V(G)|$, $\omega = \omega(G)$, $w' = w(G)$. We shall prove that PrExt can be solved in Cn^K steps for each precoloring of G , where $C = C(k, w)$ and $K = K(k, w)$ are independent of n .

We begin the precoloring extension algorithm with testing if $\omega \leq k$. For k fixed, this procedure takes $O(n^k)$ time. If $\omega > k$, then the answer to PrExt is 'no'. Therefore, from now on, we assume that $\omega \leq k$. Moreover, for technical reasons, we test in $O(n^2)$ time whether in the precoloring there are two adjacent vertices of the same color. The importance of this—seemingly superfluous—step will become clear later. Those preliminary investigations require at most $\frac{1}{4}Cn^K$ steps (when C and K are chosen to be sufficiently large.)

Let G be an intersection graph of H' , for some subdivision H' of H . By Proposition 6.5, $w(H') = w(H) \leq w$, and then a result of Scheffler [18] implies that $w' < \omega(w + 1) \leq kw + k$. Thus, the $O(n^{kw+k+1})$ algorithm of Arnborg, Corneil and Proskurowski [2] verifies that $w(G) \leq kw + k - 1$ and finds a tree decomposition $(\{X_i: i \in I\}, T = (I, F))$ of G with $\max |X_i| = w' + 1$ in polynomial time.

Define a *separation* of G as a pair (G', G'') of subgraphs $G' = (V', E')$ and $G'' = (V'', E'')$ such that $V' \cup V'' = V$, $E' \cup E'' = E$, and $E' \cap E'' = \emptyset$. For any subset $Q \subseteq V$ and positive integer m , such a separation is called (Q, m) -balanced if

$$|V' \cap V''| \leq m \quad \text{and} \quad |(V' - V'') \cap Q|, |(V'' - V') \cap Q| \leq \frac{2}{3}|Q - (V' \cap V'')|.$$

It follows from results of Robertson and Seymour [17] that G has a $(V, kw + k)$ -balanced separation of G , and it can be found in polynomial time.

Now we apply induction on n to prove the upper bound of Cn^K on the time complexity of PrExt. We assume that this bound is valid for $n \leq 12k(w+1)$ and that the $(V, kw+k)$ -balanced separation mentioned above can be constructed in at most $\frac{1}{4}Cn^K$ steps. (These requirements can be satisfied by a suitable choice of C and K .)

On the subgraph induced by $V' \cap V''$ there are at most k^{kw+k} partial extensions of the precoloring. Since this number is just a constant, we may assume that all of those partial extensions can be generated in less than $\frac{1}{4}(\frac{4}{3})^K$ steps. At this point we do not consider adjacencies of those $kw+k$ vertices to $V - (V' \cap V'')$, i.e., the partial extensions need *not* be *proper* precolorings. (Otherwise the time complexity of this step could increase to $O(n)$.)

Since there is no edge from $V' - V''$ to $V'' - V'$, the problem of PrExt on G is now 'separated' into two independent subproblems, namely PrExt on $G' = G_{V'}$ and PrExt on $G'' = G_{V''}$, for each partial extension on $V' \cap V''$. Certainly, $w(G') \leq w'$ and $w(G'') \leq w'$. Moreover, $|V'| < \frac{3}{4}n$ and $|V''| < \frac{3}{4}n$ since the separation is balanced and $n > 12k(w+1)$. Consequently, for each partial extension, the algorithm terminates in at most $2C(\frac{3}{4}n)^K$ steps, by the induction hypothesis.

We have obtained that the preliminary check and finding a balanced separation takes at most $\frac{1}{2}Cn^K$ time (together), and then the algorithm branches into at most $\frac{1}{4}(\frac{4}{3})^K$ subalgorithms taking at most $2C(\frac{3}{4}n)^K$ steps each. Thus the total running time is bounded above by

$$\frac{1}{2}Cn^K + \frac{1}{4}(\frac{4}{3})^K 2C(\frac{3}{4}n)^K = Cn^K.$$

This completes the proof of the theorem. $\square$

We note that the time bound obtained in the previous proof can be improved considerably.

Note added in proof. Problem 6.2 has been answered in the affirmative by P. Scheffler.

Acknowledgements

We would like to express our thanks to A. Frank, A. Gyárfás, J. Lehel, and L. Lovász for useful conversations. We are also grateful to P. Scheffler and the referee for their comments on the manuscript.

References

- [1] S. Arnborg, Efficient algorithms for combinatorial problems with bounded decomposability—A survey, BIT 25 (1985) 2–23.

- [2] S. Arnborg, D. G. Corniel and A. Proskurowski, Complexity of finding embeddings in a k -tree, *SIAM J. Algebraic Discrete Methods* 8 (1987) 277–284.
- [3] D. Bienstock, On embedding graphs in trees, *J. Combin Theory Ser. B*, to appear.
- [4] D. Bisztray and A. Frank, Oral communication, 1990.
- [5] B. Bollobás and A.J. Harris, List-colourings of graphs, *Graphs Combin.* 1 (1985) 115–127.
- [6] M.R. Garey, D.S. Johnson and H.C. So, An application of graph to printed circuit testing, *IEEE Trans. on Circuits and Systems* 23 (1976).
- [7] M.R. Garey, D.S. Johnson, G.L. Miller and C.H. Papadimitriou, The complexity of coloring circular arcs and chords, *SIAM J. Algebraic Discrete Methods* 1 (1980) 216–222.
- [8] M.C. Golumbic, *Algorithmic Graph Theory and Perfect Graphs* (Academic Press, New York, 1980).
- [9] A. Hilton and P. Johnson, A variation of Ryser's theorem and a necessary condition for the list-coloring problem, in: R. Nelson and R.J. Wilson, eds., *Graph colorings* (Longman, New York, 1990).
- [10] M. Hujter and Zs. Tuza, Precoloring extension. II. Graph classes related to bipartite graphs, in preparation.
- [11] M. Hujter and Zs. Tuza, Precoloring extension. III. Classes of perfect graphs, in preparation.
- [12] M. Hujter and Zs. Tuza, Precoloring extension. IV. General bounds and list colorings, in preparation.
- [13] J. Kahn, Asymptotically good list-colorings, to appear.
- [14] B. Manvel, Extremely greedy coloring algorithms, in: F. Harary and J.S. Maybee, eds., *Graphs and Combinatorics* (Wiley, New York, 1985) 257–270.
- [15] O. Marcotte and P.D. Seymour, Extending an edge-coloring, *J. Graph Theory* 14 (1990) 565–573.
- [16] F.P. Preparata and M. Sarrafzadeh, Channel routing of nets bounded degree, in: P. Bertolazzi and F. Luccio, eds., *VLSI: Algorithms and Architectures* (North-Holland, Amsterdam, 1985) 189–203.
- [17] N. Robertson and P. Seymour, Graph minors. II. Algorithmic aspects of tree-width, *J. Algorithms* 7 (1986) 309–322.
- [18] P. Scheffler, What graphs have bounded tree-width?, *Rostock. Math. Kolloq.* 41 (1990) 31–38.
- [19] P. Scheffler, Algorithms for graphs of bounded tree-width [Extended abstract], *Proc. of the International Conference "Discrete Mathematics"* (H. Sachs., ed., Eisenach, GDR, 1990) 51–54.
- [20] A. Tucker, An efficient test for circular arc graphs, *SIAM J. Comput.* 9 (1980) 1–24.
- [21] V.G. Vizing, Critical graphs with given chromatic class (in Russian), *Diskret Analiz.* 5 (1965) 9–17.