Acta Math. Univ. Comenianae
Vol. LXII, 2(1993), pp. 139–153

PRECOLORING EXTENSION WITH FIXED COLOR BOUND

J. KRATOCHVÍL

Abstract. Precoloring Extension (shortly PrExt) is the following problem: Given a graph G with some precolored vertices and a color bound k, can the precoloring of G be extended to a proper coloring of all vertices of G using not more than k colors? Answering an open problem from [**6**], we prove that PrExt with fixed color bound $k = 3$ is NP-complete for bipartite (and even planar) graphs, and we prove a general result on parametrized PrExt. We also give a simplified argument why PrExt with fixed color bound is solvable in polynomial time for graphs of bounded treewidth (and hence also for chordal graphs).

1. Introduction and Statement of the Results

All graphs considered are finite, undirected and without loops or multiple edges. A coloring of a graph is any mapping from its vertex set into a set of colors, a coloring is proper if adjacent vertices are mapped onto distinct colors. The following decision problem is introduced in [**1**] and studied in [**6, 7, 8**]:

Precoloring Extension (shortly **PrExt**)
Instance: A positive integer k and a graph G some of whose vertices are precolored using at most k colors.
Question: Can the precoloring of G be extended to a proper coloring of G using at most k colors?

Obviously, PrExt is at least as difficult as ordinary Coloring, and therefore NP-complete for general input. It is proved in [**6**] that (1) PrExt is solvable in polynomial time for split graphs, (2) NP-complete for general bipartite graphs, and again polynomially solvable for (3) complements of bipartite graphs and (4) for bipartite graphs with no induced path of length four. In [**1**], it is proved that (5) PrExt is NP-complete even for interval (and thus also for chordal) graphs.

If the color bound k is fixed, the following results were known: (6) PrExt is polynomially solvable for $k = 2$ and arbitrary G [**6**], and (7) for any fixed k for graphs of bounded treewidth [**8**]. Since 3-Colorability is a subproblem of PrExt with $k = 3$, PrExt with $k = 3$ is NP-complete. Hujter and Tuza ask in [**6**,

Received November 23, 1992; revised August 4, 1993.
1980 *Mathematics Subject Classification* (1991 *Revision*). Primary 05C15, 05C85; Secondary 68Q15, 68R10.

Problem 2.5] whether PrExt with fixed color bound is polynomially solvable on bipartite graphs. We prove

Theorem 1. *The problem PrExt is NP-complete for planar bipartite graphs even with fixed color bound* $k = 3$.

They also ask in [**6**, Problem 4.7] whether there is an integer t such that PrExt is NP-complete on P_t–free bipartite graphs. We answer this problem in affirmative even for fixed color bound:

Theorem 2. *The problem PrExt with fixed color bound* $k = 5$ *is NP-complete for bipartite graphs which do not contain induced paths of length* ≥ 13.

Theorem 1 naturally invokes the question of parametrizing PrExt by the color bound versus the number of colors used for the given precoloring. Let us consider the following problem:

(r, s, k)-PrExt
Instance: An r-partite graph G, some of whose vertices are precolored using at most s colors.
Question: Can the precoloring of G be extended to a proper coloring of G which uses at most k colors?

Theorem 1 becomes a core of the following result:

Theorem 3. *Let* r, s, k *be nonnegative integers. The* (r, s, k)*-PrExt problem is*

1. *trivial (every instance is infeasible) if* $k < s$*;*
2. *trivial (every instance is feasible) if* $k \geq r + s$ *or* $r = 1$ *and* $k \geq s$*;*
3. *polynomially solvable if* $k \leq 2$ *and* $s \leq k$*;*
4. *NP-complete in the remaining cases, i.e. when* $\max\{3, s\} \leq k \leq r + s - 1$ *and* $r > 1$.

Theorem 1 is proved in the next section and Theorem 2 in Section 3. Note that Theorem 3, which is proved in Section 4 does not depend on whether the r-partition of G is given, or G is just promised to be r-partite. This is, however, not the case if one considers the search variant of PrExt, which approach is touched in Section 5. In the last section, we give a simplified argument why PrExt with fixed color bound is polynomially solvable for graphs of bounded treewidth, and we prove that for such graphs PrExt is polynomial even if the color bound is a part of the input.

2. Planar Bipartite Graphs

We shall prove Theorem 1 in this section. We show a reduction from Planar 1-in-3 Satisfiability, a problem which is proven to be NP-complete in [**10**]:

Instance: A formula Φ with a set C of clauses over a set X of variables in conjunctive normal form such that

1. every clause contains exactly 3 distinct variables;
2. the graph $G_\Phi = (X \cup C, \{xc \mid (x \in c \in C) \vee (\neg x \in c \in C)\}$ is planar.

Question: Is there a truth assignment to the variables such that every clause receives exactly one TRUE literal?

Suppose we are given a formula $\Phi = \bigwedge_{i=1}^{n} c_i$ as an instance of Planar 1-in-3 Satisfiability, where $c_i = (l_{i1} \vee l_{i2} \vee l_{i3})$ and each literal l_{ij} is either a positive or a negated variable. We fix a planar drawing D_Φ of G_Φ and construct a graph $G(\Phi)$ by local replacements in D_Φ as follows:

1. Each variable $x \in X$ is replaced by a so called variable gadget G_x which is formed by vertices A_x, B_x joined by an edge. The vertex B_x is precolored by color 3, A_x is precolorless.
2. Each clause c_i is replaced by a so called clause gadget G_i, where

$$V(G_i) = \{C_j(i), D_j(i), E_j(i), F_j(i) \mid j = 1, 2, 3\} \cup \{C(i)\}$$

and

$$E(G_i) = \{C_j(i)E_j(i), D_j(i)F_j(i), C_j(i)C(i), C_j(i)D_j(i), C_j(i)D_{j+1}(i) \mid j = 1, 2, 3\}$$

(the addition in subscripts is modulo 3). The vertices $E_j(i), F_j(i)$ are precolored as follows

$$\phi(E_j(i)) = j,\ \phi(F_j(i)) = j + 2,\ j = 1, 2, 3$$

(addition is modulo 3), other vertices of G_i are precolorless (cf. Fig. 1).

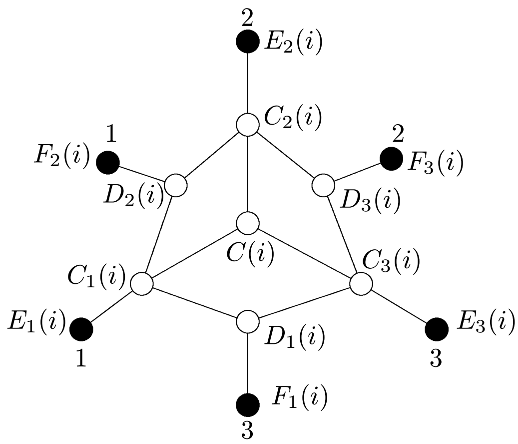


Figure 1. The clause gadget.

3. To link the variable gadgets to the clause gadgets, we use so called **linking gadgets**. Let A, B be (not necessarily distinct) vertices of a precolored graph H and let x, y, u, v be colors of $\{1,2,3\}$. Then H is called an **$A(x,y)B(u,v)$-link** if
   (a) H admits 3-colorings ϕ_1, ϕ_2 which properly extend its precoloring with (*) $\phi_1(A) = x$ and $\phi_1(B) = u$ and (**) $\phi_2(A) = y$ and $\phi_2(B) = v$;
   (b) no 3-coloring ϕ of H properly extends its precoloring unless ϕ satisfies (*) or (**).

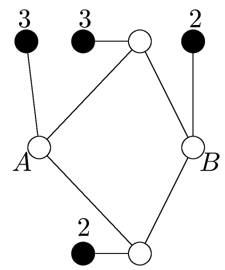


Figure 2. An $A(1,2)B(1,3)$-link.

Every variable gadget is an $A_x(1,2)A_x(1,2)$-link, and this is the simplest link possible. An $A(1,2)B(1,3)$-link is depicted in Figure 2. Concatenating such links one can construct $A(1,2)B(x,y)$-links for any $x, y \in \{1,2,3\}$, $x \neq y$, which are depicted in Figure 3. Each such link is bipartite and planar, and admits a planar drawing such that the vertices A, B are on the boundary of the outerface.

Given a variable x which occurs in a clause c_i, we define a linking gadget G_{xc_i} as follows

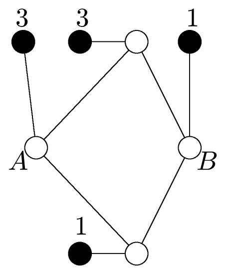
$$G_{xc_i} \text{ is a } \begin{cases} A_x(1,2)C_1(i)(2,3)\text{-link} & \text{if } l_{i1} = x, \\ A_x(1,2)C_1(i)(3,2)\text{-link} & \text{if } l_{i1} = \neg x, \\ A_x(1,2)C_2(i)(3,1)\text{-link} & \text{if } l_{i2} = x, \\ A_x(1,2)C_2(i)(1,3)\text{-link} & \text{if } l_{i2} = \neg x, \\ A_x(1,2)C_3(i)(1,2)\text{-link} & \text{if } l_{i3} = x, \\ A_x(1,2)C_3(i)(2,1)\text{-link} & \text{if } l_{i3} = \neg x, \end{cases}$$

where each linking gadget G_{xc_i} is assumed to be disjoint from all other gadgets, except for the vertices A_x, $C_j(i)$.

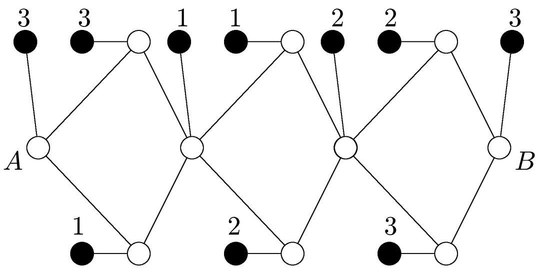
4. Set

$$G(\Phi) = \bigcup_{x\in X} G_x \cup \bigcup_{i=1}^{n} G_i \cup \bigcup_{\substack{x\in c_i \text{ or } \neg x\in c_i \\ 1\le i\le n}} G_{xc_i}.$$

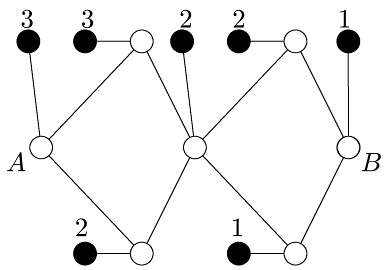
Each vertex of $G(\Phi)$ is precolored or precolorless in accord with its precoloring status in the corresponding gadget(s).



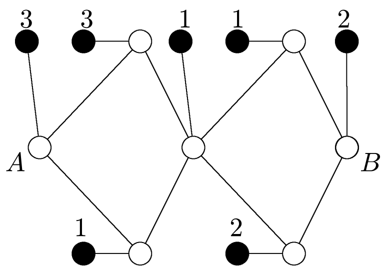
The $A(1,2)B(3,2)$-linking gadget.



The $A(1,2)B(2,1)$-linking gadget.



The $A(1,2)B(2,3)$-linking gadget.



The $A(1,2)B(3,1)$-linking gadget.

Figure 3. The linking gadgets.

Claim 1. The graph $G(\Phi)$ is planar and bipartite.

A planar drawing of $G(\Phi)$ can be obtained from the drawing D_Φ. The biparts of $G(\Phi)$ are $\{A_x, C_j(i), F_j(i) \mid x \in X,\ i = 1,2,\ldots,n,\ j = 1,2,3\}$ plus the particular vertices of the linking gadgets, and $\{B_x, D_j(i), E_j(i), C(i) \mid x \in X,\ i = 1,2,\ldots,n,\ j = 1,2,3\}$ plus the particular vertices of the linking gadgets.

Claim 2. Suppose ϕ is a proper coloring of $G(\Phi)$ which extends its precoloring. Define a truth valuation by

$$f(x) = \begin{cases} \text{TRUE} & \text{if } \phi(A_x) = 1 \\ \text{FALSE} & \text{if } \phi(A_x) = 2. \end{cases}$$

If x occurs in c_i, say $l_{ij} = x$ resp. $l_{ij} = \neg x$, then

$$l_{ij} \text{ is } \begin{cases} \text{TRUE} \\ \text{FALSE} \end{cases} \text{ in } c_i \text{ iff } \begin{cases} \phi(C_j(i)) = j+1 \\ \phi(C_j(i)) = j+2 \end{cases}$$

(addition modulo 3).

If the literals l_{i1}, l_{i2}, l_{i3} in a clause c_i were all three TRUE (or all three FALSE), it would be $\{\phi(C_1(i)), \phi(C_2(i)), \phi(C_3(i))\} = \{1,2,3\}$ and the middle vertex $C(i)$

could not be properly colored. Also if exactly two literals, say l_{ij}, l_{ij+1} were TRUE, it would be $\{\phi(C_j(i)), \phi(C_{j+1}(i)), \phi(F_{j+1}(i))\} = \{j+1, j+2, j+3\}$ and the vertex $D_{j+1}(i)$ could not be properly colored. It follows that in each clause, exactly one variable receives the value TRUE.

Claim 3. If f is a truth assignment to the variables such that in every clause, exactly one variable receives the value TRUE, the precoloring can be extended to a proper coloring of $G(\Phi)$.

Again, for every variable x, we define

$$\phi(A_x) = \begin{cases} 1 & \text{if } f(x) = \text{TRUE} \\ 2 & \text{if } f(x) = \text{FALSE.} \end{cases}$$

It follows from the definition of linking gadgets that ϕ is unique on the links, and hence again

$$\phi(C_j(i)) = \begin{cases} j+1 \\ j+2 \end{cases} \quad \text{iff} \quad l_{ij} \text{ is } \begin{cases} \text{TRUE} \\ \text{FALSE} \end{cases} \text{in } c_i$$

(addition modulo 3), if x occurs in c_i and $l_{ij} = x$ or $l_{ij} = \neg x$.

If a clause c_i contains exactly one TRUE literal, say l_{ij}, then e.g. $\phi(D_j(i)) = j$, $\phi(D_{j+1}(i)) = j+2$, $\phi(D_{j+2}(i)) = j$, $\phi(C(i)) = j+2$ is a proper extension of the precoloring inside the clause gadget.

Thus $G(\Phi)$ admits a proper coloring extension if and only if Φ is 1-in-3 satisfiable.

3. Graphs without Long Induced Paths

Hujter and Tuza proved that PrExt (with the color bound being a part of the input) is solvable in polynomial time for bipartite graphs which do not contain induced paths on 5 or more vertices. They ask in [**6**, Problem 4.7] whether PrExt is NP-complete on P_t–free bipartite graphs for some fixed t (we denote by P_t the path of length t, i.e. the path on $t+1$ vertices). Our Theorem 2 which will be proved in this section is even stronger — we prove the NP-completeness result even for relatively small fixed color bound $k = 5$. The strategy is very similar to that in the previous section. To simplify the proof, we do not impose the planarity restriction on the input graphs. Thus we may use Not-All-Equal 3-Satisfiability, a problem which is well known to be NP-complete for general graphs (but solvable in polynomial time for planar inputs) [**5**]. (An instance of Not-All-Equal 3-Satisfiability is a formula with 3 literals per clause, and it is feasible if it allows a truth assignment such that every clause contains at least one TRUE and least one FALSE literal.)

Proof of Theorem 2. Suppose we are given a formula $\Phi = \bigwedge_{i=1}^{n} c_i$ as an instance of Not-All-Equal 3-Satisfiability, where $c_i = (l_{i1} \vee l_{i2} \vee l_{i3})$ and each literal l_{ij} is either a positive or a negated variable. We construct a graph $G(\Phi)$ as follows:

1. Each variable v is replaced by a variable gadget G_v which is formed by a single precolorless vertex A_v.
2. Each clause c_i is replaced by a so called clause gadget G_i, where

$$V(G_i) = \{C_j(i) \mid j = 1,2,3\} \cup \{C(i)\} \cup \{E(i), F(i)\}$$

and

$$E(G_i) = \{C(i)E(i), C(i)F(i)\} \cup \{C_j(i)C(i) \mid j = 1,2,3\}.$$

The vertex $E(i)$ is precolored by color 1 and $F(i)$ is precolored by color 2 (thus forcing $C(i)$ to be colored by 3, 4 or 5 in any proper extension) (cf. Fig. 4).

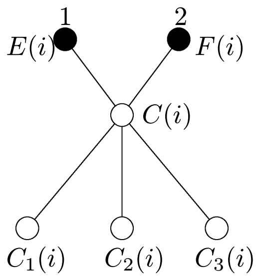


Figure 4. The clause gadget.

3. To link the variable gadgets to the clause gadgets, we use linking gadgets again. An $A(1,2)B(x,y)$-link is depicted in Figure 5, and here $\{x,y,z\} = \{3,4,5\}$. Given a variable v which occurs in a clause c_i, we define a linking gadget G_{vc_i} as follows

$$G_{vc_i} \text{ is a } \begin{cases} A_v(1,2)C_1(i)(3,4)\text{-link} & \text{if } l_{i1} = v, \\ A_v(1,2)C_1(i)(4,3)\text{-link} & \text{if } l_{i1} = \neg v, \\ A_v(1,2)C_2(i)(4,5)\text{-link} & \text{if } l_{i2} = v, \\ A_v(1,2)C_2(i)(5,4)\text{-link} & \text{if } l_{i2} = \neg v, \\ A_v(1,2)C_3(i)(5,3)\text{-link} & \text{if } l_{i3} = v, \\ A_v(1,2)C_3(i)(3,5)\text{-link} & \text{if } l_{i3} = \neg v, \end{cases}$$

where each linking gadget G_{vc_i} is assumed to be disjoint with all other gadgets, except for the vertices A_v, $C_j(i)$.

4. Denote by $G(\Phi)$ the graph with vertex set

$$V(G(\Phi)) = \bigcup_{i=1}^{n} V(G_i) \cup \bigcup_{\substack{v \in c_i \text{ or } \neg v \in c_i \\ 1 \le i \le n}} V(G_{vc_i})$$

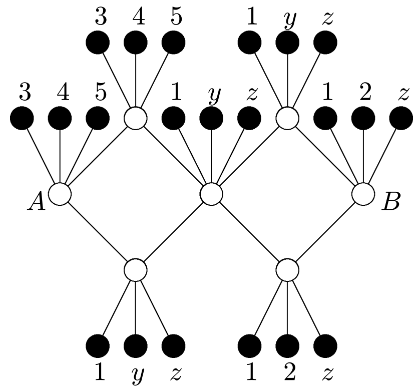


Figure 5. An $A(1,2)B(x,y)$-link.

and edge set

$$E(G(\Phi)) = \bigcup_{i=1}^{n} E(G_i) \cup \bigcup_{\substack{v \in c_i \text{ or } \neg v \in c_i \\ 1 \le i \le n}} E(G_{vc_i}) \cup \{A_v C(i) \mid v \in X,\ 1 \le i \le n\}.$$

Each vertex of $G(\Phi)$ is precolored or precolorless in accord with its precoloring status in the corresponding gadget. The egdes $A_v C(i)$ prevent $G(\Phi)$ from having long induced paths.

Claim 1. The graph $G(\Phi)$ is bipartite and P_{13}–free. Any longest induced path contains at most two vertices of type $C(i)$ and at most two variable vertices A_v. Any path longest among those which have no A_v vertex contains at most one $C(i)$ vertex and hence has length (= number of edges) at most 10. Proceeding in the discussion on the number of A_v and $C(i)$ vertices occurring in an induced path, one easily sees that any longest path involves one A_v and two $C(i)$ vertices (or vice versa) and has length 12.

Claim 2. Suppose ϕ is a proper coloring of $G(\Phi)$ which extends its precoloring. Define a truth valuation by

$$f(v) = \begin{cases} \text{TRUE} & \text{if } \phi(A_v) = 1 \\ \text{FALSE} & \text{if } \phi(A_v) = 2. \end{cases}$$

If v occurs in c_i, say $l_{ij} = v$ resp. $l_{ij} = \neg v$, then

$$l_{ij} \text{ is } \begin{cases} \text{TRUE} \\ \text{FALSE} \end{cases} \text{ in } c_i \quad \text{iff} \quad \begin{cases} \phi(C_j(i)) = j+2 \\ \phi(C_j(i)) = j+3 \end{cases}$$

(addition modulo 3). If all literals in a clause c_i were TRUE (or all FALSE), the vertices $C_1(i)$, $C_2(i)$, $C_3(i)$ would obtain together all three colors 3, 4, 5, and there would be no way to extend the coloring to the vertex $C(i)$. It follows that each clause gets at least one TRUE and at least one FALSE literal.

Claim 3. If, on the other hand, c_i has at least one TRUE and at least one FALSE literal, $\{\phi(C_1(i)), \phi(C_2(i)), \phi(C_3(i))\} \neq \{3, 4, 5\}$ and the middle vertex $C(i)$ can be colored properly. Thus, if f is a truth assignment to the variables such that there is no clause in which all literals receive the same value, one can construct a coloring extension ϕ along the lines above.

Thus $G(\Phi)$ admits a proper coloring extension if and only if Φ is not-all-equal satisfiable.

4. The Proof of Theorem 3

We will prove Theorem 3 in this section, see an illustrative Figure 6 for better orientation in the theorem. Most of the work was actually done in the preceding section, as in the current terminology, Theorem 1 reads that (2, 3, 3)-PrExt is NP-complete. Then we have the following lemma:

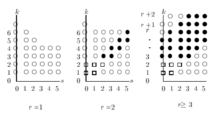


Figure 6. $\circ$ = trivial; $\square$ = polynomial; $\bullet$ = NP-complete.

Lemma 4.1. 1. *For every r, s, k, (r, s, k)-PrExt $\propto$ $(r + 1, s, k)$-PrExt.*
2. *For every $r > 1, s, k$, (r, s, k)-PrExt $\propto$ $(r, s + 1, k + 1)$-PrExt.*
3. *$(r - 1)$-Colorability $\propto$ $(r, 0, r - 1)$-PrExt for every $r \geq 4$.*
4. *3-Colorability $\propto$ $(3, 1, 3)$-PrExt.*
5. *$(2, 2, 3)$-PrExt is NP-complete.*

Proof. 1. Every r-partite graph is $(r + 1)$-partite.

2. Given an r-partite graph G with some vertices precolored by at most s colors, say $1, \dots, s$, construct a graph G' by pending a new extra vertex of degree one to each vertex of G. The new vertices are all precolored by color $k+1$. Then the coloring of G' can be completed using $k+1$ colors $1, \dots, k+1$ iff the precoloring of G can be extended using k colors $1, \dots, k$.
3. Given a graph G as an instance of $(r-1)$-Colorability (which is well known to be NP-complete for $r \geq 4$), construct a graph H in two steps as follows. First set $G' = G \times K_{r-1}$ (this is the so called direct product of graphs, G' has vertices $(u,i), u \in V(G), 1 \leq i \leq r-1$ and edges $(u,i)(v,j)$ for $uv \in E(G)$ and $i \neq j$). It is well known that G' is uniquely $(r-1)$-colorable iff G is not colorable by $r-1$ colors. Then pick a vertex of G, say u, and add a new vertex 0 to G', together with edges $0(u,i)$, $i = 1, 2, \dots, r-1$. The graph H obtained in this way is obviously r-colorable, and it is $(r-1)$-colorable iff G is $(r-1)$-colorable. Note that we can construct an r-coloring of H in polynomial time, that is, being given an r-coloring of a graph does not help in deciding its $(r-1)$-colorability.
4. Here the reduction is similar. Being given a graph G, construct $H = G \times K_3$, pick a vertex u of G and precolor the vertices $(u,1)$, $(u,2)$, $(u,3)$ of H by color 1. If G is not 3-colorable, then H is uniquely 3-colorable with color classes $V_i = \{(u,i) \mid u \in V(G)\}, i = 1,2,3$, and hence the precoloring of H is not extendable. On the other hand, any 3-coloring of G determines a 3-coloring of H such that each triple $\{(u,1),(u,2),(u,3)\}$, $u \in V(G)$ is monochromatic. Thus the precoloring of H is extendable, provided $\chi(G) \leq 3$.
5. We show $(2,3,3)$-PrExt $\propto$ $(2,2,3)$-PrExt. Given G with some vertices precolored by colors 1, 2, 3, denote by C_3 the set of vertices precolored by color 3. Pend two new vertices of degree one to each vertex of C_3, precolor one of them by color 1 and the other one by color 2, and forget the precoloring on C_3. In any proper extension of the precoloring of this new graph G', the vertices of C_3 can only receive color 3. Hence G' is a feasible instance of $(2,2,3)$-PrExt iff G is a feasible instance of $(2,3,3)$-PrExt, while the latter is NP-complete by Theorem 1. □

Proof of Theorem 3.

1. is obvious. 2. Suppose there exists an r-partition $V(G) = \bigcup_{i=1}^{r} V_i$ of G into independent sets. Denote by C the set of the precolorless vertices of G and color the vertices of $V_i \cap C$ by color $s+i$ for $i = 1, 2, \dots, r$. This is a proper extension of the given precoloring which uses exactly $r+s$ colors. The statement about $r = 1$ is obvious.
3. is proved in [**1**], a slightly different argument is as follows: Consider variables $x_u, u \in V(G)$ and set $x_u =$ TRUE iff u is colored by color 1. Now extendability of a precoloring can be expressed as an instance of 2-Satisfiability — for a vertex u precolored by 1 (resp. 2) introduce a clause (x_u) (resp. $(\neg x_u)$), and for every edge uv, introduce clauses $(x_u \vee \neg x_v)$ and $(\neg x_u \vee x_v)$.

4. This is the core of the theorem. The proof by induction on s and k hinges on Lemma 4.1.1-2:
   (4a) If $k = s$, then $(2,3,3)$-PrExt $\propto$ $(2,s,s)$-PrExt $\propto$ (r,s,s)-PrExt for $k = s \geq 3$ and $r \geq 2$. The $(2,3,3)$-PrExt problem is NP-complete by Theorem 1 and hence (r,s,k)-PrExt is NP-complete as well.
   (4b) If $k = s+1$, then $(2,2,3)$-PrExt $\propto$ $(2,s,s+1)$-PrExt $\propto$ (r,s,k)-PrExt for $k = s+1 \geq 3$ and $r \geq 2$. The $(2,2,3)$-PrExt problem is NP-complete by Lemma 4.1.5 and hence (r,s,k)-PrExt is NP-complete as well.
   (4c) If $k = s+2$, then $(3,1,3)$-PrExt $\propto$ $(3,s,s+2)$-PrExt $\propto$ (r,s,k)-PrExt for $k = s+2 \geq 3$ and $r \geq 3$. (Note that r must be greater than two, since $r = 2$ would yield $k = s + r$, the case dealt with in part 2.) The $(3,1,3)$-PrExt problem is NP-complete by Lemma 4.1.4 and hence (r,s,k)-PrExt is NP-complete as well.
   (4d) If $k \geq s+3$, then $(k-s+1,0,k-s)$-PrExt $\propto$ $(k-s+1,s,k)$-PrExt $\propto$ (r,s,k)-PrExt for $k \geq s+3 \geq 3$ and $r \geq k-s+1 \geq 4$. The $(k-s+1,0,k-s)$-PrExt problem is NP-complete by Lemma 4.1.3 and hence (r,s,k)-PrExt is NP-complete as well. □

5. The Search Version

We have noted that Theorem 3 does not change if one considers the input with a given r-coloring, or with just promised existence of one. This is not true for the following search version:

(r,s,k)-Search PrExt
Instance: A graph G which is promised to be r-partite, some of its vertices are precolored using at most s colors.
Task: Find a coloring of G which uses at most k colors and which properly extends the given precoloring (if such an extension exists).

Of course, if we considered G given with an r-partition, the result on (r,s,k)-Search PrExt would simply copy Theorem 3 (replace "trivial" by "polynomial" for search). However, the promise version is richer and we have only partial results. First it is fairly well known that knowing that a given graph is r-colorable doesn't make it easier to r-color the graph. This means, that for every $r \geq 3$, the $(r,0,r)$-Search PrExt problem is NP-hard.

Proposition 5.1. *For every $r \geq 3$ and $k \geq s+r$, the problems (r,s,k)-Search PrExt and $(r,0,k-s)$-Search PrExt are polynomially equivalent.*

Proof. (1) First we prove $(r,s-1,k-1)$-Search PrExt $\propto$ (r,s,k)-Search PrExt. Given G as an instance of $(r,s-1,k-1)$-Search PrExt, pend a new vertex of degree one to each precolorless vertex, and precolor the pending vertices by color k. Any extension of this precoloring induces a proper coloring of G by $k-1$ colors. By

induction on s, it follows that $(r, 0, k-s)$-Search PrExt $\propto$ (r, s, k)-Search PrExt.
(2) Next we show (r, s, k)-Search PrExt $\propto$ $(r, 0, k-s)$-Search PrExt. Given a graph G as an instance of (r, s, k)-Search PrExt, delete all precolored vertices. Since $k - s \geq r$, the resulting graph G' is $(k-s)$-colorable, and any $(k-s)$-coloring which uses other colors than the precoloring is a proper extension. □

Thus we can rephrase Theorem 3 as follows:

Theorem 4. *Let r, s, k be nonnegative integers. The (r, s, k)-Search PrExt problem is*

1. *trivial (every instance is infeasible) if $k < s$ or $k < r$;*
2. *polynomial if $k \leq 2$ and $\max\{r, s\} \leq k$;*
3. *polynomial if $r = 1$ and $k \geq s$;*
4. *polynomial if $r = 2$ and $k \geq s + 2$;*
5. *NP-hard if $r = 2$ and $\max\{3, s\} \leq k \leq s + 1$;*
6. *NP-hard if $r \geq 3$ and $\max\{3, s\} \leq k \leq r + s$;*
7. *for every $r \geq 3$ either*
   (7a) *there exists a positive constant M_r such that (r, s, k)-Search PrExt is NP-hard for $r + s \leq k < r + s + M_r$ and polynomial for $k \geq r + s + M_r$, or*
   (7b) *(r, s, k)-Search PrExt is NP-hard for every $k \geq r + s$.*

A simple observation shows that if (7a) holds for some r and $r + 1$ then $M_r \leq M_{r+1}$. Hence either (7a) holds for every r, or there is a constant $R \geq 3$ such that (7a) holds for every r s.t. $3 \leq r < R$, and (7b) holds for every $r \geq R$. We conjecture that $R = 3$, i.e. that (7b) holds for all $r \geq 3$. Note that in particular for $s = 0$, this conjecture implies that chromatic number does not admit a polynomial approximation, in a very strong way. Recent results based on alternative description of the class NP form a breakthrough in this direction. It follows from the result of Linial [private communication] that $M_3 > 1$. However, it is still open whether $M_r = \infty$ for some r.

6. Graphs of Bounded Treewidth

We noted that PrExt with fixed color bound is solvable in polynomial time for graphs of bounded treewidth. (Actually a slightly more general result is proved in [**8**].) We note here that the result on bounded treewidth follows from the method of monadic second-order logic for graphs, developed by Courcelle [**4**]. The metaresult is that every graph property expressible in the monadic second-order logic is decidable in polynomial time on graphs of bounded treewidth. In the case of PrExt with color bound k, we consider unary predicates

$$\mathbf{lab}_i(v) = \begin{cases} \text{TRUE} & \text{if } v \text{ is precolored by color } i \\ \text{FALSE} & \text{otherwise,} \end{cases}$$

for $1 \le i \le k$, and

$$\mathbf{lab}_0(v) = \begin{cases} \text{TRUE} & \text{if } v \text{ is not precolored} \\ \text{FALSE} & \text{otherwise.} \end{cases}$$

A binary predicate $\mathbf{edg}(u, v)$ expresses whether u and v are adjacent.

Given a graph G with some precolored vertices, the following formula written in the monadic second-order logic expresses that the precoloring can be extended to the entire G:

$$\exists M_1, M_2 \dots, M_k (\forall v (\bigvee_{i=1}^{k} (v \in M_i) \wedge \bigwedge_{i=1}^{k} (\mathbf{lab}_i(v) \Rightarrow v \in M_i))$$
$$\wedge \forall u, v (\mathbf{edg}(u, v) \Rightarrow \bigwedge_{i=1}^{k} \neg (u \in M_i \wedge v \in M_i))).$$

Note that this implies that PrExt with fixed color bound is polynomial on chordal graphs. If $\omega(G) \ge k$ (where $\omega(G)$ denotes the size of a maximum clique of a chordal graph G), then the answer is à priori “no”. Otherwise, $w(G) = \omega(G) - 1 \le k - 1$ and we face an instance of PrExt with fixed color bound and bounded treewidth.

In [**8**, Problem 6.2], Hujter and Tuza ask whether PrExt is polynomially solvable even if k is a part of the input. This is indeed so:

Theorem 5. *The problem PrExt restricted to graphs of treewidth $\le w$ is solvable in time $O(k^{w+1}n)$, where k is the color bound and n is the number of vertices of the input graph.*

Proof. Let T, $\{X_t \mid t \in V(T)\}$ be a tree decomposition of G of width w, i.e. T is a tree (and we may suppose that it is binary), each X_t is a subset of $V(G)$ of size at most $w + 1$ such that (1) for every $u \in V(G)$, the set $V_u = \{t \mid u \in X_t\}$ induces a connected subgraph of T and (2) for every edge $uv \in E(G)$, there is a $t \in V(T)$ such that $u, v \in X_t$. A minimal tree decomposition has size linear in n. To simplify the technical details below, we assume that $|X_t| = w + 1$ for all t (this may be achieved by introducing dummy vertices to G if necessary). Note that due to a recent result of Bodlaender [**2**], given a graph of treewidth w one can find a tree decomposition of width w in linear time.

Choose a vertex t_0 of T as a root and call a vertex t a predecessor of s if s is the first vertex on the (unique) path from t to t_0 in T. For each t, denote by T_t the subtree induced by t, its predecessors, the predecessors of its predecessors etc. Similarly, denote by G_t the subgraph of G induced by $\bigcup_{s \in V(T_t)} X_s$.

The idea of the algorithm is to keep track of all feasible extensions on G_t from leaves to the root. For every $t \in V(T)$, $X \subset X_t$ and every coloring $\phi_X : X \to$

$\{1, 2, \ldots, k\}$, we set

$$A_{t,X,\phi_X} = \begin{cases} 1 & \text{if } G_t \text{ admits an extension } \phi \text{ of the precoloring} \\ & \text{such that } \phi|_X = \phi_X \\ 0 & \text{otherwise.} \end{cases}$$

Then obviously, G admits an extension iff $A_{t_0,\emptyset} = 1$. The point is that A can be computed in polynomial time. First of all, there are $|V(T)| \cdot (k+1)^{|X_t|}$ triples $(t, X \subset X_t, \phi_X)$, and thus A requires at most $O(n)(k+1)^{w+1}$ bits.

The evaluation of A goes from the leaves of T to their successors. At each leaf l, we simply check all k^{w+1} possible total colorings of X_l, if they properly extend the precoloring. For each feasible one (i.e., for each ϕ_{X_l} such that $A_{l,X_l,\phi_{X_l}} = 1$) we set $A_{l,X,\phi_X} = 1$ for each $X \subset X_l$ and $\phi_X = \phi_{X_l}|_X$. Processing a leaf requires time $O((2k)^{w+1})$.

For an internal vertex t, we again check the total colorings of X_t first. For a coloring $\phi\colon X_t \to \{1, 2, \ldots, k\}$ we check whether it properly extends the coloring on X_t (in constant time) and then we check each predecessor s of t whether $A_{s,X_t \cap X_s,\phi|_{X_t \cap X_s}} = 1$. We set $A_{t,X_t,\phi} = 1$ if and only if this holds for each predecessor s. Then for each feasible total coloring of X_t, we augment the list of feasible partial colorings of X_t similarly as we did above for the leaves. Each internal vertex is thus processed in time $O(2(2k)^{w+1}) = O(k^{w+1})$. □

Added in Proof. Precoloring extension and its application to scheduling problems was also considered by Jansen et al. in [**9**], [**3**]. They also prove that PrExt with fixed color bound is NP-complete for bipartite graphs, but their proof does not apply to planar bipartite graphs.

Acknowledgment. The author thanks the referee for comments that led to improvement of the presentation.

References

1. Biró M., Hujter M. and Tuza Z., *Precoloring extension. I. Interval graphs*, Discr. Math. **100** (1992), 267–279.

2. Bodlaender H. L.:, *A linear time algorithm for finding tree-decompositions of small treewidth*, Tech. report RUU-CS-92-27, Utrecht University.

3. Bodlaender H. L., Jansen K. and Woeginger G., *Scheduling of incompatible jobs*, Workshop on Graph Theoretical Concepts in Computer Science 1992, Lecture Notes in Computer Sci. 657, Springer Verlag, Berlin, 1992, pp. 37–49.

4. Courcelle B., *The monadic second order logic of graphs I: Recognizable sets of finite graphs*, Information and Comput. **85** (1990), 12–75.

5. Garey M. R. and Johnson D. S., *Computers and Intractability: A Guide to the Thoery of NP-completeness*, H. Freeman, New York, 1979.

6. Hujter M. and Tuza Z., *Precoloring extension. II. Graph classes related to bipartite graphs*, (submitted).

7. ———, *Precoloring extension. III. Classes of perfect graphs*, (submitted).

8. ———, *Precoloring extension. IV. Partial k-trees and list colorings*, (submitted).

9. Jansen K. and Scheffler P., *Some coloring results for tree like graphs*, Workshop on Graph Theoretical Concepts in Computer Science 1992, Lecture Notes in Computer Sci. 657, Springer Verlag, Berlin, 1992, pp. 50–59.

10. Mansfield A., *Determining the thickness of graphs is NP-hard*, Proc. Math. Cambridge Phil. Soc. **93** (1983), 9–23.

J. Kratochvíl, Charles University, Prague, Czech Republic