13

Coloring Graphs On-line

HAL A. KIERSTEAD

1 Introduction

This chapter presents a survey of three types of results concerning on-line graph coloring: The first type deals with the problem of on-line coloring k-chromatic graphs on n vertices, for fixed k and large n. The second type concerns fixed classes of graphs whose on-line chromatic number can be bounded in terms of their clique number. Examples of such classes include interval graphs and the class of graphs that do not induce a particular radius two tree. The last type deals with classes of graphs for which First-Fit performs reasonably well in comparison to the best on-line algorithms. Examples of such classes include interval graphs, the class of graphs that do not induce the path on five vertices, and d-degenerate graphs.

An *on-line graph (digraph)* is a structure $G^{\prec} = (V, E, \prec)$, where $G = (V, E)$ is a graph (digraph) and $\prec$ is a linear order on V. (Here V will always be finite.) The ordering $\prec$ is called an *input sequence.* Let $G_n^{\prec}$ denote the on-line graph induced by the $\prec$-first n elements $V_n = \{v_1 \prec \cdots \prec v_n\}$ of V. An algorithm A that properly colors the vertices of the on-line graph $G^{\prec}$ is said to be an *on-line coloring algorithm* if the color of the n-th vertex v_n is determined solely by the isomorphism type of $G_n^{\prec}$. Intuitively, the algorithm A colors the vertices of G one vertex at a time in the externally determined order $v_1 \prec \cdots \prec v_n$, and at the time a color is irrevocably assigned to v_n, the algorithm can only see G_n. For example, the on-line coloring algorithm First-Fit (FF) colors the vertices of $G^{\prec}$ with an initial sequence of the colors $1, 2, \ldots$ by assigning the vertex v the least color that has not already been assigned to any vertex adjacent to v. The number of colors that an algorithm A uses to color $G^{\prec}$ is denoted by $\chi_A(G^{\prec})$. For a graph G the maximum of $\chi_A(G^{\prec})$ over all input sequences $\prec$ is denoted by $\chi_A(G)$. If Γ is a class of graphs, the maximum of $\chi_A(G)$ over all G in Γ is denoted by $\chi_A(\Gamma)$. The on-line chromatic number of Γ, denoted by $\chi_{\rm ol}(\Gamma)$, is the minimum over all on-line algorithms A of $\chi_A(\Gamma)$.

For a graph $G = (V, E)$, the chromatic number, clique number, and independence number of G are denoted by $\chi(G)$, $\omega(G)$, and $\alpha(G)$. Let u and v be vertices in G. If u and v are adjacent, we may write $u \sim v$. Let $N(v) = \{w \in V : v \sim w\}$ and $d(v) = |N(v)|$. If $G^{\prec} = (V, E, \prec)$ is an on-line graph, then $N^{\prec}(v) = \{w \in V : v \sim w \text{ and } w \prec v\}$ and $d^{\prec}(v) = |N^{\prec}(v)|$. If G is isomorphic to H we may write $G \approx H$. The set $\{1, 2, \ldots, n\}$ is denoted by $[n]$. For a sequence $\sigma = (\sigma_1, \ldots, \sigma_n)$ a subsequence of the form $\sigma = (\sigma_1, \ldots, \sigma_i)$ is called an *initial* sequence of σ and a subsequence of the form $(\sigma_i, \ldots, \sigma_n)$ is called a *final* sequence of σ. Let $|\sigma|$ be the length of σ.

Our goal is to find on-line coloring algorithms that perform well on various classes of graphs. To see what this might mean, we begin by considering some simple examples. In later sections we explore in more detail the issues raised by these examples. We will include some illustrative proofs.

Example 1. (Gyárfás and Lehel [9]).
For every positive integer k there exists a tree T_k on 2^{k-1} vertices such that for every on-line coloring algorithm A, $\chi_A(T_k) \geq k$.

Proof. We begin by defining the tree T_k. Let $D = \{\sigma : \sigma \text{ is a strictly decreasing sequence of positive integers}\}$. For $\sigma \in D$, let $V_\sigma = \{\tau \in D : \sigma \text{ is an initial segment of } \tau\}$. Let T_σ be the tree on the vertex set V_σ such that τ is adjacent to τ' iff $|\tau| + 1 = |\tau'|$ and $\tau' \in V_\tau$ or vice versa. We shall call σ the root of T_σ and abbreviate $T_{(t_1,\ldots,t_i)}$ by $T_{t_1,\ldots,t_i}$. In particular $T_k = T_{(k)}$. Note that if τ is a final segment of σ, then there exists an isomorphism from T_τ to T_σ that maps $\tau^\wedge\rho$ to $\sigma^\wedge\rho$. So $T_k - (k) = T_{k,1} + \ldots + T_{k,k-1}$. Putting (k) back, we see that (1) $T_k \approx T_{k-1} + T_{k-1} + e$, where e is an edge joining the roots of the two copies of T_{k-1}, and the root of $T_{k-1} + T_{k-1} + e$ can be either one of the endpoints of e. In particular, $|V_k| = 2^{k-1}$.

Let $S_{k,i} = T_{k,1} + \ldots + T_{k,i}$. The key property of T_k that we exploit is that (2) for any $i < k$, there exists an embedding of $S_{k,i}$ into T_k that maps (k, i) to (k) and is extendible to an automorphism of T_k. It follows that an on-line algorithm that has only seen a subgraph isomorphic to $S_{k,i}$ cannot distinguish between (k, i) and (k). Property (2) is easily proved by induction on $k - i$. The base step $k - i = 1$ follows immediately from (1). The induction step follows from the induction hypothesis applied to the pair $\{k - 1, i\}$ and from the base step.

Let P_k be a partial ordering on $S_{k,k-1}$ such that $x\,P_k\,y$ iff $x \in T_{k,i}$, $y \in T_{k,j}$, and $i < j$. We claim that for every positive integer k and on-line coloring algorithm A, there exists a total ordering $\prec_k$ of $S_{k,k-1}$ such that $\prec_k$ extends P_k and A assigns each of the vertices $(k, 1), \ldots, (k, k-1)$ a distinct color when $S_{k,k-1}$ is presented in the order $\prec_k$. It then follows that if (k) is presented last, then A uses a k-th color to color (k). Arguing inductively, assume that we have shown this for the case $k = m$ and consider the case $k = m+1$. Since $S_{k,m-1} \approx S_{m,m-1}$, there exists an ordering $\prec_m$ such that when A is applied to $S_{k,m-1}$ in the order

$\prec_m$, A uses distinct colors on the set $Q = \{(k,1), \ldots, (k, m-1)\}$. Let B be an on-line algorithm that colors $T_{k,m}$ in the same way that A colors $T_{k,m} \approx T_m$ after first coloring $S_{k,m-1}$ in the order $\prec_m$. By the induction hypothesis applied to B, instead of A, there exists an ordering $\prec^*$ of $T_{k,m}$ extending the preimage of P_m, such that when A is applied to $S_{k,m}$ in the order $\prec_m + \prec^*$, A uses distinct colors on the set $R = \{(k,m,1), \ldots, (k,m,m-1)\}$. If A uses the same colors on both Q and R, then (k,m) gets a new color and we are done. Otherwise some root (k,m,i) gets a new color α. Because $\prec^*$ extends the preimage of P_m, at the time (k,m,i) is colored by the algorithm B has only seen a subgraph of $T_{k,m} \approx T_m$ which is isomorphic to $S_{m,i}$. So by (2) we can reorder $V_{k,m}$ so that (k,m) looks like $(k,m,m-1)$ and is colored with α by A. □

This example shows that we cannot bound the on-line chromatic number of a graph solely in terms of its chromatic number, even in the case of trees. In Section 2 we will obtain non-trivial bounds on the on-line chromatic number of graphs on n vertices in terms of their chromatic number and n. Because of the following example, our emphasis will be on fixed k and large n.

Example 2. (Szegedy [34]).
For every on-line algorithm A and positive integer k, there exists an on-line graph $G^{\prec}$ on n vertices such that $\chi(G^{\prec}) \leq k$, $n \leq k2^k$, and $\chi_A(G^{\prec}) \geq 2^k - 1$.

Proof. We construct $G^{\prec}$ in stages; $G_s^{\prec}$ is constructed at the s-th stage, which consists of three steps. First we introduce a new vertex v_s together with all edges from v_s to previous vertices. Next we determine the color $A(v_s) = c$, that A assigns v_s. We may assume that $c \in \{c_1, \ldots, c_{2^k-1}\}$. Finally we assign a color $f(v_s) \in \{r_1, \ldots, r_k\}$ to v_s. Let C_j be the set of vertices that A has colored c_j, R_i be the set of vertices that we have colored r_i, and $X_{ij} = R_i \cap C_j$. We shall try to maintain the following induction hypothesis:

(1) f is a proper k-coloring and
(2) $|X_{ij}| \leq 1$, for all $i \in [k]$ and $j \in [2^k - 1]$.

By (1) $G^{\prec}$ is k-colorable and by (2) $G^{\prec}$ has less than $k2^k$ vertices. Thus it suffices to show that we can maintain (1) and (2) until A uses $2^k - 1$ colors.

Let $S \subset [k]$. We say that S is *represented* if there exists j such that $i \in S$ iff $X_{ij} \neq \emptyset$. If every non-empty subset of $[k]$ is represented then A has already used $2^k - 1$ colors and we are done. Otherwise, suppose S is a non-empty subset of $[k]$ which is not represented. Let v_s be adjacent to v iff $v \in R_i$ and $i \notin S$. Suppose A colors v_s with c_j. Then $X_{ij} = \emptyset$ for all $i \notin S$. Thus, since S is neither empty nor represented, there exists $i \in S$ such that $X_{ij} = \emptyset$. Let $f(v_s) = i$. □

After the first two negative examples one might wonder whether there are any interesting on-line coloring algorithms. Our next example is a simple on-line coloring algorithm with a nontrivial performance bound.

Example 3. (Kierstead [13]).
For every positive integer n, there exists an on-line algorithm B such that

$\chi_B(G) \leq 2n^{1/2}$, for any graph G on n vertices that contains neither C_3 nor C_5.

Proof. Consider the input sequence $v_1 \prec v_2 \prec \cdots \prec v_n$ of an on-line graph $G^{\prec}$ that contains neither C_3 nor C_5. Initialize by setting $W_i = \emptyset$ for all $i \in \lceil 2n^{1/2} \rceil - \lceil n^{1/2} \rceil$. At the s-th stage the algorithm processes the vertex v_s as follows.

1. If there exists $i \in \lceil 2n^{1/2} \rceil$ such that v_s is not adjacent to any vertex colored i, then let j be the least such i and color v_s with j.
2. Otherwise, if there exists $i > n^{1/2}$ such that $v_s \in N(W_i)$, then let j be the least such i and color v_s with j.
3. Otherwise, let j be the least integer $i > n^{1/2}$ such that $W_i = \emptyset$. Set $W_j = \{v \in N^{\prec}(v_s) : \text{the color of } v \text{ is at most } n^{1/2}\}$ and color v_s with color j. (Note that $|W_j| \geq n^{1/2}$, since Case 1 does not hold. Also, for all $i < j$, $W_i \cap W_j = \emptyset$, since Case 2 does not hold.)

Suppose for a contradiction that two adjacent vertices x and y, with $x \prec y$, have the same color j. Clearly y is not colored by Step 1. Thus $j > n^{1/2}$, and thus x is not colored by Step 1. Since only the first vertex colored j can be colored by Step 3, y must be colored by step 2. If x is colored by Step 3, then $W_j \subset N(x)$ and $y \in N(W_j)$, and so x and y have a common neighbor in W_j. But then G contains C_3, a contradiction. If x is colored by Step 2, then both x and y are in $N(W_j)$ and so either they have a common neighbor in W_j, and we are done as before, or they have distinct neighbors in W_j, each of which is adjacent to the first vertex colored j. In this case G contains C_5, a contradiction. So B produces a proper coloring. At most $n^{1/2}$ colors are used in Step 1. Since the W_j are disjoint and have size at least $n^{1/2}$, at most $n^{1/2}$ colors are used for Steps 2 and 3 combined. Thus $\chi_B(G^{\prec}) \leq 2n^{1/2}$. □

In Section 3 we study special classes of graphs that have the property that the on-line chromatic number of any graph in the class can be bounded in terms of its chromatic number, in fact even in terms of its clique size. The following was probably the first such result.

Example 4. (Kierstead and Trotter [25]).
There exists an on-line coloring algorithm A such that for any interval graph G, $\chi_A(G) \leq 3\omega(G) - 2$; moreover for any on-line coloring algorithm A and any positive integer k, there exists an interval graph G such that $\omega(G) = k$ and $\chi_A(G) \geq 3k - 2$.

Proof. We shall only prove the first statement. First we prove by induction that for all k there exists an on-line algorithm A_k such that if $G^{\prec}$ is an on-line interval graph with $\omega(G) = k$ then $\chi_{A_k}(G^{\prec}) \leq 3\omega(G) - 2$. The base step $k = 1$ is trivial, so consider the induction step $k > 1$. Consider the input sequence $v_1 \prec \ldots \prec v_n$ of G. The algorithm A_k will maintain an on-line partition of V. When a new vertex v_s is presented, A_k puts v_s into a set of B, if $\omega(B \cup \{v_s\}) < k$; otherwise

A_k puts v_s into C. If v_s is put into B, it is colored by A_{k-1} applied to B using the set of colors $[3k-5]$; otherwise v_s is colored by First-Fit applied to C using colors greater than $3k-5$. It suffices to show that First-Fit uses at most 3 colors on C. To prove this, we will show that the maximum degree of C is at most 2. For each vertex x of G, let I_x be the interval that corresponds to x in some interval representation of G. If $x \in C$, then x is in a k-clique K such that $K - \{x\} \subset B$. Let p_x be a point in the intersection of all intervals corresponding to vertices in K. Note that $p_x \notin I_y$ for any other vertex $y \in C$, since otherwise $K \cup \{y\}$ would be a $(k+1)$-clique in G. Suppose for a contradiction that x is adjacent to three vertices in C. Without loss of generality we can assume that for two of them, say y and z, $p_x < p_y < p_z$. Then $I_y \subset I_x \cup I_z$, since I_x intersects I_z and I_y is contained in the interval from p_x to p_z. But then $p_y \in I_x \cup I_z$, which is a contradiction.

The algorithm A guesses that $\omega(G) \leq k$ and uses A_k to color G until a vertex v_s is presented that forms a $(k+1)$-clique. At this time the algorithm switches to A_{k+1}. This does not cost any colors because A_{k+1} would have also used A_k to color the first $s-1$ vertices anyway. □

In Section 4, we shall study classes of graphs for which First-Fit performs well. The next two examples show that the class of trees has this property, but the class of 2-colorable graphs certainly does not.

Example 5. (Gyárfás and Lehel [9]).
For any tree T, $\chi_{\text{ol}}(T) = \chi_{\text{FF}}(T)$.

Proof. Notice that the maximum degree of the tree T_k constructed in Example 1 is $k-1$. Thus $\chi_{\text{FF}}(T_k) \leq k$, and so by Example 1, $\chi_{\text{FF}}(T) = k$. We shall show by induction on k, that for any tree T, if First-Fit colors a vertex of v of T with color k, then T contains a copy of T_k with root v. It follows that First-Fit is an optimal on-line coloring algorithm for trees. The base step is trivial, so consider the induction step. If First-Fit colors v with $k+1$, then for all positive integers $i \leq k$, v is adjacent to a vertex v_i that First-Fit has colored i. By the induction hypothesis, v_i is a root of a copy U_i of T_i in $T - v$. Since T is acyclic, distinct U_i are in distinct components of $T - v$. It follows that $\{v\} \cup \bigcup_{i \leq k} U_i$ is a copy of T_{k+1}. □

Example 6. For every positive integer n there exists a 2-colorable graph G on n vertices such that $\chi_{\text{FF}}(G) = n/2$.

Proof. Let $B_n = K_{n,n} - M$, where $M = \{a_i b_i : i \in [n]\}$, is a perfect matching in $K_{n,n}$. Let $\prec$ be the input sequence $(a_1, b_1, a_2, b_2, \ldots, a_n, b_n)$. Then First-Fit colors each a_i and b_i with the color i. □

2 Performance bounds for general graphs

In this section we consider the problem of finding good on-line coloring algorithms for the class of all graphs. Let $\phi(k,n)$ be the least integer $t\,(\leq n)$

for which there exists an on-line algorithm A such that $\chi_A(G) \leq t$, for any k-colorable graph G on n vertices. We have already seen in Example 2 that $\phi\left(k, k2^k\right) \geq 2^k - 1$. Here we shall be interested in the case where k is fixed and n is much larger than $k2^k$. In the definition of ϕ, the algorithm A is allowed to depend on n. In other words, the algorithm knows the number of vertices of G ahead of time as in Example 3. This makes the statement of some algorithms simpler, but does not change the order of ϕ as the following doubling technique shows.

Lemma 1. *Let Γ be a class of graphs and g be an integer valued function on the positive integers such that $g(x) \leq g(x+1) \leq g(x)+1$, for all x. If for every n, there exists an on-line coloring algorithm A_n such that for every graph $G \in \Gamma$ on n vertices, $\chi_{A_n}(G) \leq g(n)$, then there exists a fixed on-line coloring algorithm A such that for every $G \in \Gamma$ on n vertices, $\chi_A(G) \leq 4g(n)$.*

Proof. Choose a sequence of integers $c_0 = 1, c_1, c_2, \ldots$ such that $2g\left(c_i\right) = g\left(c_{i+1}\right)$. Color the first c_0 vertices using A_{c_0}, then color the next c_1 vertices, using A_{c_1} and a new palette, then color the next c_2 vertices, using A_{c_2} and a new palette, etc. This algorithm will color every graph $G \in \Gamma$ on $\sum_{0 \leq h \leq i} c_h$ vertices, with at most $2g(c_i)$ colors. To see this, argue by induction on i. The base step $i=0$ is trivial, so consider the induction step $i = j+1$. We use at most $2g(c_j) = g\left(c_i\right)$ colors on the first $\sum_{0 \leq h \leq j} c_h$ vertices by the induction hypothesis, and at most $g\left(c_i\right)$ colors on the last c_i vertices. So we use at most $2g\left(c_i\right)$ colors in all.

Now suppose that $G \in \Gamma$ is a graph on n vertices with $\sum_{0 \leq h < i} c_h \leq n < \sum_{0 \leq h \leq i} c_h$. After coloring $\sum_{0 \leq h < i} c_h$ vertices we guess that there are going to be $\sum_{0 \leq h \leq i} c_h$ vertices, which we will be able to color with the allotted number of colors, because by the claim we have accumulated a surplus of $2g\left(c_{i-1}\right) = g\left(c_i\right)$ colors. Thus we use at most $4g\left(c_{i-1}\right) \leq 4g(n)$ colors. □

We begin our study of ϕ with the case $k=2$. In Example 1 we saw that $\phi(2,n) \leq \lg n$. The next theorem shows that $\phi(2,n) = \Theta(\log n)$, a quite satisfactory answer.

Theorem 2. (Lovász, Saks, and Trotter [27]).
There exists an on-line algorithm A such that for every on-line 2-colorable graph G on n vertices, $\chi_A(G) \leq 2\lg n$.

Proof. Consider the input sequence $v_1 \prec v_2 \prec \cdots \prec v_n$ of an on-line 2-colorable graph $G^{\prec}$. When v_i is presented there is a unique partition (I_1, I_2) of the connected component of $G_i^{\prec}$ to which v_i belongs, into independent sets such that $v_i \in I_1$. The algorithm A assigns v_i the least color not already assigned to some vertex of I_2.

It suffices to show that if A uses at least t colors on any connected component of $G_i^{\prec}$, then that connected component contains at least $2^{\lfloor t/2 \rfloor}$ vertices. We argue by induction on i and note that the base step is trivial. For the induction step, observe that if A assigns v_i color $k+2$, then A must already have assigned color k to some vertex $v_p \in I_2$ and color $k+1$ to some other vertex in I_2. Thus A must have assigned color k to some vertex $v_q \in I_1$. Since A assigned v_p and v_q the same color, v_p and v_q must be in separate components of $G_r^{\prec}$, where $r = \max\{p, q\}$. Thus by the induction hypothesis, each of these connected components must have at least $2^{\lfloor k/2 \rfloor}$ vertices and so the component of v_i in $G_i^{\prec}$ has at least $2^{\lfloor (k+2)/2 \rfloor}$ vertices. □

The situation is not nearly as clear for $k > 2$. Vishwanathan generalized the lower bound in the case $k = 2$, showing that $\phi(k, n) = \Omega\left(\log^{k-1} n\right)$.

Theorem 3. (Vishwanathan [36]).
For all integers k and n, $\phi(k, n) \geq (\lg n/(4k))^{k-1}$.

Proof. In order to simplify calculations, we will prove the weaker result that for every k, there exists $\varepsilon_k > 0$ such that for all n, $\phi(k, n) \geq \varepsilon_k (\lg n)^{k-1}$. The key idea of the proof is to show that there exists a function $f(k, n)$ satisfying the initial conditions $f(2, n) \geq \varepsilon_2 \log(n)$, $f(k, k) = k$, and recurrence relation $f(k+1, 3n) = f(k+1, n) + \frac{1}{2} f(k, n)$, such that for every on-line algorithm A, there exists an on-line k-colorable graph $G^{\prec}$ on n vertices and a proper k-coloring c of G such that A uses at least $f(k, n)$ colors on some color class of c. It then follows that $\phi(k, n) \geq f(k, n) \geq \varepsilon_k (\log n)^{k-1}$, for the some constant $\varepsilon_k > 0$.

We argue by double induction on k and then n. Fix an on-line algorithm A. Using Example 1, the base steps follow easily. We shall construct a $(k+1)$-colorable on-line graph $G^{\prec}$ on $3n$ vertices and a proper $(k+1)$-coloring c^* of G such that A uses at least $f(k, n) + \frac{1}{2} f(k, n)$ colors on some color class of c^*. By the secondary induction hypothesis there exists a $(k+1)$-colorable on-line graph $X^{\prec}$ on n vertices and a proper $(k+1)$-coloring c of X such that A uses at least $f(k+1, n)$ colors on some color class I of c. Let A' be the on-line algorithm that colors an on-line graph $H^{\prec}$ in the same way that A would color H after first coloring a disjoint copy of $X^{\prec}$. Then again using the secondary induction hypothesis, there exists an on-line $(k+1)$-colorable graph $Y^{\prec}$ on n vertices so that $Y^{\prec}$ is disjoint from $X^{\prec}$ and there exists a proper $(k+1)$-coloring c' of Y such that A' uses at least $f(k+1, n)$ colors on some color class I' of c'. Then when A is presented with $X^{\prec}$ followed by $Y^{\prec}$, A uses a set C of at least $f(k+1, n)$ colors on I and another set D of at least $f(k+1, n)$ colors on I'. If $|C \cup D| \geq f(k+1, n) + \frac{1}{2} f(k, n)$, then we are done since $I \cup I'$ is a color class of the $(k+1)$-coloring $c^* = c \cup c'$. Otherwise, $|C \cap D| \geq f(k+1, n) - \frac{1}{2} f(k, n)$. Let A'' be the on-line algorithm that colors an on-line graph $H^{\prec}$ in the same way that A would after first coloring a disjoint copy of $X^{\prec}$ followed by a disjoint copy of $Y^{\prec}$, if every vertex in H were adjacent to every vertex in I. By the primary induction hypothesis there exists a k-colorable on-line graph $Z^{\prec}$ on n vertices and a proper k-coloring c'' of Z such that A'' uses at least $f(k, n)$ colors on some color class I'' of c''. Let $G^{\prec}$ be $X^{\prec}$ followed by $Y^{\prec}$ followed by $Z^{\prec}$ together with all possible edges from Z to I. Then none of the colors A uses on I'' are used on I, and so A uses at least $f(k+1, n) + \frac{1}{2} f(k, n)$ colors on $I' \cup I''$. Since G has a proper $(k+1)$-coloring c^* with a color class containing $I' \cup I''$, we are done. □

Until recently, the best upper bound on $\phi(k,n)$ was given by the following theorem, where $\lg^{(k)}$ is the lg function iterated k times.

Theorem 4. (Lovász, Saks, and Trotter [27]).
There exists an on-line algorithm A such that for every k-colorable on-line graph $G^{\prec}$ on n vertices, $\chi_A(G^{\prec}) = O\left(n \lg^{(2k-3)} n / \lg^{(2k-4)} n\right)$. □

Their proof made use the following combinatorial lemma, whose proof follows easily from Inclusion-Exclusion or Lovász [26].

Lemma 5. *Let n be a positive integer and let δ be a positive real less than one. If F is a family of subsets of $[n]$ such that for all distinct $D, E \in F$, $|E| \geq \delta n$ and $|D \cap E| < \delta^2 n/2$, then $|F| < 2/\delta$.* □

Very recently the author used the same lemma to obtain a $O\left(n^{1-1/k!}\right)$ upper bound on $\phi(k,n)$.

Theorem 6. (Kierstead [13]).
For every positive integer k, there exists an on-line algorithm A_k and an integer N such that, for every on-line k-colorable graph $G^{\prec}$ on $n \geq N$ vertices, $\chi_{A_k}(G^{\prec}) \leq n^{1-1/k!}$.

Proof. To avoid messy calculations, we shall prove a somewhat weaker statement, but the full strength of the theorem can be obtained from the proof we give by being a little more careful with the details and initial conditions. We shall show that for all positive integers k there exist positive constants C and ε such that for all positive integers n there exists an on-line coloring algorithm $A_{k,n}$ such that for all k-colorable graphs G on n vertices $\chi_{A_{k,n}}(G) \leq Cn^{1-\varepsilon}$. We argue by induction on k. The base step is trivial; for the induction step assume that we have proved that there exist positive constants C and ε such that for all $i \leq k$, there exists an on-line algorithm $A_{i,n}$ such that $\chi_{A_{i,n}}(G) \leq Cn^{1-\varepsilon}$, for all i-colorable graphs G on n vertices.

Fix n. We shall describe $A = A_{k+1,n}$ in terms of two parameters α and δ, where $0 < \alpha, \delta < 1$. For $i \leq k$, let $A_i = A_{i,n'}$, where $n' = n^{\alpha}$. Set $\delta_i = 2^{-i}\delta$, $s_0 = n$, and $s_i = \delta_{i-1}s_{i-1}$, for $i \leq k$. Later we shall apply Lemma 5 with $n = s_i$ and $\delta = \delta_i$. Let $G^{\prec} = (V, E, \prec)$ be an on-line $(k+1)$-colorable graph on n vertices, and let Z be a subset of V.

First we describe a dynamic data structure in terms of the life cycle of the strange mythical species of *witnesses*. *Male* witnesses are *witness vertices* in $V - Z$. *Female* witnesses are certain *witness sets* contained in Z. A *witness tree* records the female genealogy of witnesses starting from the original witness set Z (Eve). From time to time witness sets will *spawn* large *litters* of *daughters*. Each of the daughters in a litter is a subset of her *mother*. Each daughter D in the litter has a distinct (!) *father* $F(D)$, who is a witness vertex that is adjacent in G to every vertex in D. Once a witness set is spawned, it will never gain or lose elements. However Eve is special in that Eve was not spawned and will gain, but not lose, elements. The witness sets form a tree with Eve at the 0-th level, the daughters of Eve at the 1-st level, their daughters at the 2-nd level, and so on, through the k-th generation. A witness set at the i-th level is called an i-witness set. For all $i > 0$, an i-witness set has size s_i. At some times some of the witness sets may *die*. Once they die, they will never *live* again. If they never die, they are *immortal*; otherwise they are *mortal*. If all the daughters in a single litter die, then the mother also dies (of grief).

Next we describe the on-line coloring algorithm A, using the above data structure. For any i-witness set W, with $i < k$, let $N^*(W) = \{v \in V-Z : |N(v) \cap W| \geq s_{i+1}\}$. If W is a k-witness, then $N^*(W) = N(W)$. The algorithm will maintain a partition $\{S_W : W \text{ is a witness set}\}$ of $V - Z$ such that each $S_W \subset N^*(W)$. Each S_W will be partitioned by $P_W = \{X_W(j) : j \in [t_W]\}$. The last part $X_W(t_W)$ of this partition is called the *active* part. When new elements enter S_W they will be put in the active part. Let $X = \bigcup\{P_W : W \text{ is a witness set}\}$. Call $X_W(j) \in X$ *small* if it has size less than n^α. Otherwise it is *large*. The algorithm will partition V into Z, at most $n^{1-\alpha}$ large subsets of size n^α, and a bounded number of small subsets. Each of these subsets will be colored from disjoint palettes of colors. The palette for Z will have δn colors and each of the other palettes will have $Cn^{\alpha(1-1/\varepsilon)}$ colors.

Consider the input sequence $v_1 \prec v_2 \prec \cdots \prec v_n$ of $G^\prec$. At the s-th stage the algorithm processes the vertex v_s as follows.

1. If $d^\prec(v_s) < \delta n$, then put v_s in Z. Color v_s by First-Fit applied to Z, using a palette of size δn.

2.1. Otherwise v_s is a witness vertex. Find a *live* i-witness set W, with i as large as possible subject to the condition that $v_s \in N^*(W)$. Such a witness set exists by the fact that $|N(v_s) \cap Z| \geq \delta n$ and so $v_s \in N^*(Z)$, provided we can prove (Lemma 7) that Eve is immortal.

2.2. Put v_s in the active part $X_W(t)$, $t = t_W$, of P_W. Color v_s by A_i applied to $X_W(t)$, using a palette of size $Cn^{\alpha(1-\varepsilon)}$. (By step 2.3, $|X_W(t)| \leq n^\alpha$.)

2.3. If after the addition of v_s, $|X_W(t)| = n^\alpha$, then set $t_W = t+1$ and set $X_W(t_W) = \emptyset$. Then $X_W(t)$ is large.

2.4. If $n^{\alpha(1-\varepsilon)}$ colors have been used on $X_W(t)$, then we have a proof that $\chi(X_W(t)) \geq i+1$. Set $t_W = t+1$ and set $X_W(t+1) = \emptyset$. (We may have just done this.) In this case, if $i = k$, then W dies. (This may cause some female ancestors of W to die of grief.) Otherwise $i < k$ and W spawns a new litter $\{D_v : v \in X_W(t)\}$, where each daughter D_v is a s_{i+1}-subset of $N(v) \cap W$. The father of D_v is v, for each $v \in X_W(t)$.

This completes the description of the algorithm A. To show that the algorithm is well defined, we need the following Lemma.

Lemma 7. *Eve is immortal.*

Proof. Suppose that Z is mortal. Let c be a proper $(k+1)$-coloring of G. First consider any mortal $(i-1)$-witness set M, with $i \in [k]$. Since M is mortal, M has a litter L such that every daughter $D \in L$ is mortal. When L is spawned, $\chi(\{F(D) : D \in L\}) \geq i$. Thus $|\{c(F(D)) : D \in L\}| \geq i$. It follows that, setting

$W_0 = Z$, we can find a collection $\{W_i : i \in [k]\}$ such that W_i is a mortal daughter of W_{i-1} and $|\{c(F(W_i)) : i \in [k]\}| = k$. Every father in the set $\{F(W_i) : i \in [k]\}$ is an ancestor of W_k and so is adjacent to every vertex in W_k. Thus c must color every vertex in W_k with the same color. It follows that c restricted to $N(W_k)$ is a proper k-coloring. Since W_k is mortal A_k must use at least $n^{\alpha(1-\varepsilon)}$ colors on the k-colorable graph induced by $N(W_k)$, which is a contradiction. □

Clearly A produces a proper coloring of G. It remains to bound the number of colors that A uses. The key step is the next lemma that bounds the number of litters a witness set can spawn.

Lemma 8. *Every i-witness set M has less than $2/\delta_i$ litters.*

Proof. We may assume that M is alive since after M dies, M will have no more litters. Then each litter of M contains a live $(i+1)$-witness set. Suppose W and U are two live daughters of M from distinct litters. Then there exist distinct j and j' such that $F(W) \in X_M(j)$ and $F(U) \in X_M(j')$. Say $j < j'$. At the stage that $F(U)$ is processed, all the vertices in W have already been processed. Thus $|W \cap U| < s_{i+2} = \frac{\delta_i^2}{2} s_i$, since otherwise $F(U)$ would be put in S_W instead of S_M. Thus by Lemma 5, M has less than $2/\delta_i$ litters. □

Let $Q = \{X_W(j) \in X : X_W(j) \text{ is small}\}$. We claim that $|Q| \le 2^{k^2} (n^\alpha/\delta)^k$. For any i-witness set W, $|\{X_W(j) \in P_W : X_W(j) \text{ is small}\}|$ is at most one more than the number of litters of W. Note that a k-witness set spawns $l_k = 0$ litters and, by Lemma 8, an i-witness set spawns less than $l_i = 2/\delta_i$ litters. Let w_i be the number of i-witness sets. Then $w_0 = 1$ and $w_{i+1} \le w_i l_i n^\alpha$, since each of the w_i i-witness sets spawns less than l_i litters of at most n^α daughters. It follows that $w_i \le 2^{i(i+1)/2} (n^\alpha/\delta)^i$. So $|Q| \le \sum_{0 \le i \le k} w_i l_i \le 2 w_k \le 2^{k^2} (n^\alpha/\delta)^k$.

The algorithm A partitions V into at most $2^{k^2} (n^\alpha/\delta)^k$ small pieces and at most $n^{1-\alpha}$ large pieces. Each piece is colored with at most $C n^{\alpha(1-\varepsilon)}$ colors, except that Z is colored with at most δn colors. Thus, setting $\delta = n^{-\alpha\varepsilon}$ and $\alpha = 1/(k+1+k\varepsilon)$, the number of colors used by the algorithm is at most

$$\begin{aligned} \delta n + C n^{\alpha(1-\varepsilon)} \left(2^k n^\alpha/\delta\right)^k + C n^{1-\alpha} n^{\alpha(1-\varepsilon)} &= \delta n + C 2^{k^2} n^{\alpha(k+1+k\varepsilon-\varepsilon)} + C n^{1-\alpha\varepsilon} \\ &= n^{1-\alpha\varepsilon} + 2^{k^2} C n^{1-\alpha\varepsilon} + C n^{1-\alpha\varepsilon} \\ &= \left(2C + 2^{k^2}\right) n^{1-\alpha\varepsilon}. \end{aligned}$$ □

We could improve the performance of the last algorithm if we could further limit the number of witness sets. One way to do this is to improve the bounds on the size of the litters. The size of the litters of an i-witness set W is determined by the least t_i such that whenever A_i uses more than a given number c_i of colors on the active part X of P_W, X contains a subgraph on t_i vertices whose chromatic number is at least $i+1$. Clearly $t_0 = 1$, $t_1 = 2$, and by Example 3 we can take $t_2 = 5$, when $c_0 = 0$, $c_1 = 1$, and $c_2 = 2n^{1/2}$. These observations lead to the more elegant bounds in the statement of Theorem 6 and, for small values of k, the following noticeable improvements in the algorithm.

Theorem 9. (Kierstead [13]).
There exists an on-line algorithm A_3 such that for every on-line 3-colorable graph $G^{\prec}$ on n vertices, $\chi_{A_3}(G^{\prec}) < 20n^{2/3}\log^{1/3} n$.

Theorem 10. (Kierstead [13]).
There exists an on-line algorithm A_4 such that for every on-line 4-colorable graph $G^{\prec}$ on n vertices, $\chi_{A_4}(G^{\prec}) < 120n^{5/6}\log^{1/6} n$.

These algorithms are not only on-line, but also run in polynomial time. From this point of view they are quite good since the best off-line algorithms for polynomial time coloring of 3-colorable graphs use $n^{3/14}$ colors ([2]). The author and Kolossa obtained much tighter bounds in the case of perfect graphs. Let $\pi(k,n)$ be the least integer t for which there exists an on-line algorithm A such that $\chi_A(G) \le t$, for any k-colorable perfect graph G on n vertices.

Theorem 11. (Kierstead and Kolossa [19]).

$$\Omega\left(\log^{k-1} n\right) = \pi(k,n) = O\left(n^{10k/\log\log n}\right)$$

□

Example 3 illustrates the basic idea behind the proof of Theorem 11, but this idea must be iterated $\log^{(3)} n$ times and extended to graphs that do not induce any odd cycles. The actual proof is much more difficult and requires many on-line coloring techniques. The lower bound on π is derived from Vishwanathan's construction (Theorem 3). This suggests that the known lower bound on $\phi(k,n)$ is far from tight. But maybe this lower bound is close to the truth for $\pi(k,n)$. In the case of chordal graphs, Irani has an upper bound of the form $O(k \log n)$ (see Theorem 34). The following problems remain open and very interesting.

Problem 12. Find tighter bounds on $\phi(k,n)$ for fixed k and large n, especially for $k \in \{3,4,5\}$. Does $\phi(3,n) = O\left(n^{1/2}\right)$?

Problem 13. Find tighter bounds on $\pi(k,n)$ for fixed k and large n. Does there exist a function $p(k)$ such that $\pi(k,n) < \log^{p(k)} n$?

Vishwanathan studied randomized on-line algorithms. His lower bounds (Theorem 3) were actually proved for these more powerful algorithms.

Theorem 14. (Vishwanathan [36]).
There exists a randomized on-line algorithm A such that for every k-colorable on-line graph $G^{\prec}$ on n vertices, the expected value of $\chi_A(G^{\prec}) = O\left(k2^k n^{(k-2)/(k-1)}(\lg n)^{1/(k-1)}\right)$. Moreover, for any randomized on-line algorithm B, there exists a k-colorful on-line graph $G^{\prec}$ on n vertices such that the expected value of $\chi_B(G^{\prec}) = \Omega(1/(k-1)(\lg n/(12(k+1)) + 1)^{k-1})$. □

3 On-line χ-bounded classes

In this section we consider classes Γ of graphs, for which there exists an on-line algorithm A such that for all $G \in \Gamma$, $\chi_A(G)$ can be bounded by a function of $\omega(G)\,(\leq \chi(G))$, regardless of the number of vertices in G. More precisely, we say that Γ is *on-line χ-bounded* iff there exists an on-line algorithm A and a function $g(k)$, such that $\chi_A\left(G^{\prec}\right) \leq g\left(\omega(G)\right)$, for any on-line presentation $G^{\prec}$ of any $G \in \Gamma$. Similarly, Γ is *χ-bounded* if there exists a function $f(k)$ such that $\chi\left(G\right) \leq f\left(\omega(G)\right)$, for all $G \in \Gamma$. Most of the time we will not be concerned with the size of the function g; the important point is that it does not depend on the number of vertices of G.

The results of this section have their roots in the author's previous work in recursive combinatorics and a beautiful graph theoretical conjecture formulated independently by Gyárfás and Sumner. The problems the author considered in recursive combinatorics can be very roughly described as follows. Given a countably infinite graph G, design an algorithm to color each vertex v of G using only certain types of local information (in particular, only finitely much information) about v. Depending on the amount of information allowed, in increasing order, the graphs may be recursive, highly recursive, or decidable. Usually, results about coloring recursive graphs, such as Bean [1], Kierstead [14], and Kierstead and Trotter [25] translate immediately to on-line results, while results on highly recursive or decidable graphs such as Kierstead [15] [16], Manaster and Rosenstein [28], and Schmerl [31] do not. The starting point for the work of this section is Theorem 15, which we will state after introducing some more terminology.

An *on-line ordered* set is an on-line digraph $D^{\prec}$ such that D is an ordered set, i.e., D is transitive, antisymmetric, and antireflexive. The *comparability* graph of an ordered set $D = (V, A)$ is the undirected graph $G = (V, E)$, where $E = \{xy : (x, y) \in A \text{ or } (y, x) \in A\}$. Similarly the *cocomparability* graph of D is the undirected graph $G^c = (V, E^c)$, where $E^c = \{xy : (x, y) \notin A \text{ and } (y, x) \notin A \text{ and } x \neq y\}$. A *chain (antichain)* in D is an independent set in the cocomparability graph G^c (comparability graph G). The *height (width)* of an ordered set is the number of vertices in the maximum chain (antichain). Notice that the height of D is the clique size of G and the width of D is the clique size of G^c. It is well known that G (and hence G^c) is perfect.

Theorem 15. (Kierstead [14]).
There exists an on-line algorithm A that will partition the vertices of any on-line ordered set of width w into at most $\left(5^w - 1\right)/4$ chains. □

Theorem 15 does not assert the existence of an on-line coloring algorithm that will color every on-line cocomparability graph $G^{\prec}$ with $\left(5^w - 1\right)/4$ colors. The problem is that the algorithm of Theorem 15 receives as input the digraph D of an ordered set; in general this provides more information than the cocomparability graph. This was shown rigorously by Penrice [29]. The best lower bounds for Theorem 15 were obtained by Szemerédi.

Theorem 16. (Szemerédi [35]).
For every integer w and on-line chain partitioning algorithm A, there exists an on-line ordered set $D^{\prec}$ with width w such that A partitions D into at least $\binom{w+1}{2}$ chains. □

The author [14] had previously derived a super linear lower bound and shown that at least five chains were necessary in the case $w = 2$. Recently Felsner has improved the upper bound in the case $w = 2$. (This also gives a slight improvement in the general upper bound.)

Theorem 17. (Felsner [5]).
There exists an on-line algorithm A that will partition the vertices of any on-line ordered set of width 2 into at most 5 chains. □

Over the last fifteen years Theorem 17 is the only progress on the following natural problem.

Problem 18. Let $p(w)$ be the least integer such that there exists an on-line algorithm A such that A will partition the vertices of any on-line ordered set of width w into at most $p(w)$ chains. Improve the bounds $\binom{w+1}{2} \leq p(w) \leq (5^w - 1)/4$. Is $p(w)$ polynomial? □

The analogous problem for antichains is much simpler.

Theorem 19. (Schmerl [30]).
There exists an on-line algorithm that will partition any on-line ordered set of height h into at most $\binom{h+1}{2}$ antichains; moreover for every positive integer h and on-line algorithm A, there exists an on-line ordered set $D^{\prec}$ such that A cannot partition $D^{\prec}$ into fewer than $\binom{h+1}{2}$ antichains.

Proof. We only prove the upper bound. Consider the input sequence $v_1, \ldots, v_n$ of an on-line ordered set D with height h. At stage s we process the vertex v_s by putting v_s into the antichain $A_{a,b}$ where $a = a(s)$ is the number of vertices in the longest chain (at stage s) with maximum element v_s and $b = b(s)$ is the number of vertices in the longest chain (at stage s) with minimum element v_s. Then $2 \leq a + b \leq h + 1$. It follows that there are at most $\binom{h+1}{2}$ choices for the sets $A_{a,b}$. To see that the sets $A_{a,b}$ really are antichains, consider two comparable vertices v_s and v_t with $s < t$. If $v_s < v_t$ in D, then $a(s) < a(t)$. Otherwise $v_s > v_t$ in D and $b(s) < b(t)$. □

Note that the class of comparability graphs is not on-line χ-bounded since it contains the class of trees which is not on-line χ-bounded by Example 1. The following conjecture of Schmerl from 1978 motivated a lot of work, including Example 4.

Conjecture 20. (*Schmerl*).
The class of cocomparability graphs is on-line χ-bounded. □

A solution of Schmerl's Conjecture required some purely graph theoretical results. For a graph H, let Forb(H) be the class of all graphs that do not contain an induced copy of H. Quite independently of any work on on-line algorithms, and independently of each other, Gyárfás and Sumner made the following conjecture.

Conjecture 21. (Gyárfás [7], Sumner [33]).
For any tree T, Forb(T) is χ-bounded. □

The conjecture is essentially the strongest possible. It would be false if T were replaced by a graph H that contained a cycle (say of length t), since any graph with girth greater than t is in Forb(H) and Erdös and Hajnal [4] have shown that there are graphs with arbitrarily large girth and arbitrarily large chromatic number. Moreover if the conjecture is true for trees it is easy to show that it is also true for forests. Gyárfás [8] gave an easy proof to show that it is true for paths. The author and Penrice built on earlier work of Gyárfas, Szemerédi, and Tuza [11], to prove the following off-line theorem. The general conjecture is still open.

Theorem 22. (Kierstead and Penrice [20]).
For any tree T with radius at most two, Forb(T) is χ-bounded.

Gyárfás and Lehel [10] made a fundamental and unexpected breakthrough in on-line coloring when they proved the following theorem.

Theorem 23. (Gyárfás and Lehel [10]).
Forb(P_5) is on-line χ-bounded, but Forb(P_6) is not on-line χ-bounded.

Theorem 23 was the first hint of a connection between Conjectures 20 and 21. Then Gyárfás made the following observation. Let S be the radius two tree obtained by subdividing each edge of a star on four vertices. It is well known that no cocomparability graph induces S and so the class of cocomparability graphs is contained in Forb(S). Thus to prove Conjecture 20, it suffices to show that Forb(S) is χ-bounded. With this challenge, the author, Penrice and Trotter proved the following very general theorem.

Theorem 24. (Kierstead, Penrice, and Trotter [22]).
For any tree T, Forb(T) is on-line χ-bounded iff T has radius at most two.

The proof of Theorem 24 is much too long to present here. However the proof for the special case of Forb(S), which provided most of the motivation for attacking the general theorem, is considerably simpler and illustrates many of the key techniques needed to prove the general theorem.

Corollary 25. *Forb(S) is on-line χ-bounded.*

Proof. We shall first prove that Forb(S) is off-line χ-bounded and then use the proof as a basis for constructing an on-line algorithm. Let $R(\omega, \alpha)$ be the Ramsey function such that for any graph G on $R(\omega, \alpha)$ vertices either $\omega(G) \geq \omega$ or $\alpha(G) \geq \alpha$. Let f be the function on the positive integers defined inductively by:

$$f(1) = 1 \text{ and } f(\omega) = f(\omega-1)+\omega+\omega f(\omega-1)\left(\omega^2\left(R\left(\omega, R(\omega+1,3)\right)+1\right)^2+1\right)$$

We shall prove by induction on $\omega(G)$, that $\chi(G) \leq f(\omega(G))$, for every graph $G = (V, E)$ in Forb(S). The base step, $\omega = 1$, is trivial, so consider a graph $G \in$ Forb(S) with $1 < \omega(G) = \omega$.

Let $\{Q_1, \ldots, Q_t\}$ be a maximal collection of ω-cliques in G such that for all distinct $i, j \in [t], N(Q_i) \cap N(Q_j) = \emptyset$. For each $i \in [t]$, let $N_i = N(Q_i) - \bigcup_{j<i} N(Q_j)$. Let $Q = \bigcup_{j\in[t]} Q_j$, $N = \bigcup_{j\in[t]} N_j$, and $X = V - Q - N$. The ω-cliques Q_i are called *templates* and the sequence $Q_1, N_1, \ldots, Q_t, N_t$ is called a *template sequence.* Since Q is a union of disjoint ω-cliques, we can color Q with ω colors. Since $\omega(X) < \omega$, by the induction hypothesis, we can color X with $f(\omega - 1)$ new colors. These $\omega + f(\omega - 1)$ colors will not be used on any of the vertices of $V - Q - X$, so it suffices to show that $\chi(N) \leq \omega f(\omega - 1)\left(\omega 2\left(R\left(\omega, R(\omega+1,3)\right)+1\right)^2+1\right)$.

Since each $N_i \subset \bigcup_{q\in Q} N(q)$, it can be colored with $\omega f(w-1)$ colors. However, if we try to color each of the N_i with the same set of $\omega f(w - 1)$ colors, two adjacent vertices $v_i \in N_i$ and $v_j \in N_j$ may be assigned the same color. To avoid this problem each vertex of N will be assigned a two coordinate color. The first coordinate, called the *local* color and assigned as above, will insure that two adjacent vertices in the same N_i are assigned different colors. The second coordinate, called the *global* color will take care of the problem of adjacency between vertices in different N_i. Since t is unbounded, we can not simply use disjoint sets of colors for the different N_i. We shall need the following lemma, which is the only place in the proof of the theorem that we use the hypothesis that $G \in$ Forb(S).

Lemma 26. *Let* $d(s) = 2\left(R\left(\omega, R(\omega+1,3)\right)+1\right)^{s-1}$. *Then for every vertex* v *in* G, v *is connected to vertices in at most* $d(s)$ *templates by paths with at most* s *edges.*

Proof. We argue by induction on s. The base step $s = 1$ follows from the fact $G \in$ Forb(S): Suppose $v \sim q_i$, where $q_i \in Q_i$, for $i \in \{j_1 < j_2 < j_3\}$. Since each Q_i is a maximum clique there exist vertices $y_i \in Q_i$ such that not $v \sim y_i$, for $i \in \{j_1 < j_2 < j_3\}$. But then $\{v, q_i, y_i : i \in \{j_1 < j_2 < j_3\}\}$ induces S in G, which is a contradiction.

For the induction step, suppose that a vertex v is connected to vertices in $d+1$ distinct templates by paths with at most s edges, where $d = d(s)$. Chose a minimal set of vertices $F \subset N(v)$ such that each of these $d+1$ templates contains a vertex which is either connected to some vertex in F by an induced path with exactly $s-1$ edges or is connected to v by a path with at most $s-1$ edges. By the induction hypothesis $(|F|+1)\,d(s-1) \geq d+1$, and so $|F| \geq R\,(\omega, R(\omega+1,3))$. Using Ramsey's Theorem, and the fact that every vertex in F is adjacent to v, there exists an independent subset $F_0 = \{v_1, \ldots, v_p\} \subset F$ with cardinality $p = R(\omega+1,3)$. Using the minimality of F, for each $v \in F_0$ there exists a template T_i such that v_i is the only vertex in $F \cup \{v\}$ to which any vertex of T_i is connected by an induced path (say R_i) with exactly $s-1$ edges. Say $v \sim v_i \sim y_i$ in R_i. Then neither $v \sim y_j$ nor $v_i \sim y_j$, if $i \neq j$. Using Ramsey's Theorem again, there exist $j_1 < j_2 < j_3$ such that $\{y_i : i \in \{j_1, j_2, j_3\}\}$ is an independent set. But then $\{v, v_1, y_i : i \in \{j_1, j_2, j_3\}\}$ induces S in G, which is a contradiction. □

In order to assign a global color to the vertices in N, we construct an auxiliary graph B. The vertex set of B is the set of templates $\{Q_1, \ldots, Q_t\}$. Two templates Q_x and Q_y are adjacent if there is a path from one to the other with at most three edges. By Lemma 26, the maximum degree of B is at most $\omega d(3)$, since each of the ω vertices in Q_i can be connected to at most $d(3)$ templates by paths with at most three edges. Thus B can be colored with $\omega d(3)+1$ colors. The global color of a vertex $v \in N_i$ is the color of Q_i in B. To see that this gives a proper coloring, consider two adjacent vertices x and y in N. If x and y have the same local color, there exist distinct indices i and j such that $x \in N_i$ and $y \in N_j$. But then Q_i is adjacent to Q_j in B, so x and y are assigned different global colors. This completes the proof of the off-line case.

We still must show that there exists a function g and an on-line algorithm A such that $\chi(G) \leq g\,(\omega(G))$, for every graph $G \in \mathrm{Forb}(S)$. It suffices to show by induction on ω that there exist on-line algorithms A_ω, for $\omega = 1, 2, \ldots$ and a function $h(\omega)$ such that $\chi_A(G) \leq h(\omega)$, for every graph $G \in \mathrm{Forb}(S)$ such that $\omega(G) = \omega$: First guess that $\omega = 1$ and use A_1. If a 2-clique is found, guess that $\omega(G) = 2$ and start using A_2 with a new set of colors, etc. Then $g(\omega) = \sum_{j \leq \omega} h(\omega)$. The base step is trivial, so consider the induction step.

The major problem in developing an on-line algorithm from the proof of the off-line case is that we cannot possibly construct the template sequence on-line. We can only add a template to the sequence after we have seen all the vertices in the clique, but by this time we may have missed some of its neighbors. The key idea is to consider the neighbors of neighbors of vertices in the templates. A minor problem will be that we can not maintain the auxiliary graphs on-line. Our on-line algorithm $A = A_\omega$ will maintain a list of templates $Q_1, \ldots, Q_{t(s)}$, where $Q_i = \{x_{i,1} \prec \cdots \prec x_{i,\omega}\}$. A template Q_i will enter the end of the list at the time $x_{i,\omega}$ is presented. Once a template has entered the list it will not change position or leave the list. When a new vertex v_s is presented, A will assign v_s to exactly one of the sets N, L, D, or H. The sets N and D are more finely partitioned as: $N = \bigcup\{N_{i,j} : i \in \{1, \ldots, t(s)\}, j \in \{1, \ldots, \omega\}\}$ and $D = \bigcup\{D_i : i \in \{1, \ldots, t(s)\}\}$. Then each of the sets of vertices N, L, D, H, will be colored with disjoint sets of colors.

(N) If v_s is adjacent to some vertex in some template in the current template list, let i be the least such index such that $v_s \sim x_{i,k}$, for some k, and let j be the least such k. Put v_s in $N_{i,j}$.

(L) Otherwise, if v_s is in an ω-clique Q in $G - \left(N \cup \bigcup_{1 \le i \le t(s)} Q_i\right)$, then add $Q = Q_{t(s)+1}$ to the template list and put v_s in L.

(D) Otherwise, if v_s is connected to some vertex in some template in the template list by a path with two edges, let i be the *largest* index such that, for some j, v_s is connected to some $x_{i,j}$ by a path with two edges. Put v_s in D_i.

(X) Otherwise put v_s in X.

Clearly $\omega(X) < \omega$. Thus the on-line algorithm A can use the on-line algorithm $A_{\omega-1}$ to color the vertices of H with one set of $h(\omega - 1)$ colors. Also L is an independent set, so we can use one special color to color L. Thus it remains to color the vertices of N and D. As in the off-line proof, each vertex in N, and also D, will be assigned a two coordinate color. The local color insures that two adjacent vertices in $N_i = \bigcup_{1 \le j \le \omega} N_{i,j}$ or D_i are assigned different colors, while the global color insures that two adjacent vertices with the same local color are assigned different colors.

We consider the local coordinate. For N the local coordinate is assigned as in the off-line case, using the on-line algorithm $A_{\omega-1}$. In order to use $A_{\omega-1}$ to assign a local coordinate to the vertices of D_i, we must first show that $\omega(D_i) < \omega$. Suppose $Q = \{q_1 \prec \ldots \prec q_\omega\}$ is an ω-clique in D_i and consider the situation at the time q_ω was presented. Since q_ω was added to D_i instead of N, q_ω is not adjacent to any vertex in any template in the template list. Since q_ω was not added to L, some $q \in Q$ must be adjacent to some vertex in a template Q_j. Thus q_ω is connected to Q_j by a path through q with two edges. Since q is not in N, Q_j must have been added to the template list after q was presented and thus $i < j$. But then q_ω would have been assigned to D_j. We conclude that $\omega(D_i) < \omega$, for all i, and assign the local coordinate to each vertex in D using $A_{\omega-1}$.

It remains to determine the global color. First consider the vertices of N. If we could color the vertices of the auxiliary graph B on-line, we would be done. However this is not possible since the auxiliary graph is not presented on-line. Two templates Q_x and Q_y may start out being non-adjacent, but when a new vertex of $G^\prec$ is presented they may suddenly become adjacent. On the other hand, the degree of a template in B can only increase $\omega d(3, \omega)$ times. The on-line algorithm A will maintain a two coordinate, $(\omega d(3) + 1)^2$-coloring of B such that (i) the first coordinate of a template is the current degree of the template in B, (ii) the second coordinate insures that two templates which are adjacent in B and have the same degree in B are assigned different colors, and (iii) the second coordinate of a color assigned to a template will only change when the degree of the template changes. The global color of a vertex in N_i will be the color assigned to Q_i in B by A at the time the vertex is presented. To assign the global color to a vertex in D, we define another auxiliary graph B' on the templates, where two templates Q_x and Q_y are adjacent iff there is a path from a vertex in Q_x to a vertex in Q_y with at most five edges. Since each template has ω vertices, the maximum degree of B' is bounded by $\omega d(5, \omega)$. Thus as above we need only $(\omega d(5, \omega) + 1)^2$ colors for the global color of vertices in D. Thus $h(\omega)$

is defined recursively by $h(1) = 1$ and

$$h(\omega) = \omega h(\omega - 1)\,(\omega d(3,\omega) + 1)^2 + 1 + h(\omega - 1)\,(\omega d(5,\omega) + 1)^2 + h(\omega - 1)\ .\square$$

4 First-Fit χ-bounded classes

In this section we consider classes of graphs for which First-Fit performs reasonably well in comparison to the best on-line algorithm for the class. We begin by continuing our study of classes of graphs defined by forbidding certain induced subgraphs. Later we consider various classes of d-degenerate graphs such as trees, interval graphs, and chordal graphs. A First-Fit coloring of a graph $G = (V, E)$ with the colors $[t]$ produces a structure called a *wall*. A wall W in G is a partition $\{L_1, \ldots, L_t\}$ of V into independent sets such that for all vertices $v \in L_j$, there exists a vertex $u \in L_i$ such that v is adjacent to u, whenever $i < j$. The independent sets L_j are called *levels* of the wall. The *height* of W is t. It is easy to see that G contains a wall of height t iff $t \leq \chi_{\mathrm{FF}}(G)$.

A class of graphs Γ is First-Fit χ-bounded if there exists a function f such that for all graphs $G \in \Gamma$, $\chi_{\mathrm{FF}}(G) \leq f(\omega(G))$. It follows immediately from Ramsey's theorem that if S is a star, then Forb(S) is First-Fit χ-bounded. Also it is well known that any graph in Forb (P_4) is perfect, and moreover First-Fit produces an optimal coloring, where P_k is the path on k vertices. Gyárfás and Lehel [9] showed that if T is a tree that is not in Forb $(K_1 \cup K_1 \cup K_2)$, then Forb(T) is not First-Fit χ-bounded. Since P_5 is the only tree in Forb $(K_1 \cup K_1 \cup K_2)$ that is neither a star nor P_4, they asked whether Forb (P_5) was χ-bounded. This was answered affirmatively by the author, Penrice and Trotter.

Theorem 27. (Kierstead, Penrice and Trotter [23]).
The class Forb (P_5) is First-Fit χ-bounded, and thus, for any tree T, Forb(T) is First-Fit χ-bounded iff T does not contain $K_1 \cup K_1 \cup K_2$ as an induced subgraph.

Proof. We must show that there exists a function f such that for all $G \in$ Forb (P_5), $\chi(G) \leq f(\omega(G))$. We argue by induction on $\omega(G)$. The base step $\omega(G) = 1$ is trivial, so consider the induction step $\omega(G) = k$. Let $R\,(a_1, a_2, a_3)$ be the Ramsey function such that for any 3-coloring of the edges of the complete graph on $R\,(a_1, a_2, a_3)$ vertices, there exists $i \in [3]$ and a complete subgraph on a_i vertices whose edges are all colored i. The following Lemma provides the main technical tool for defining $f(k)$.

Lemma 28. *For any integer t, if a graph $G \in$ Forb (P_5) with clique size k contains a wall W of height $1 + R(f(k-1) + 1, t, 3)$, then for every vertex x in the top level of W, there exists an induced subgraph H contained in $G - x$ that has a wall W' of height t such that the top level of W' has exactly one vertex y, which is the only vertex of H adjacent to x.*

By the lemma, for sufficiently large t, we can let $f(k) = 1{+}R(f(k{-}1){+}1, t, 3)$, since if there were a graph $G \in$ Forb (P_5) with clique size k that contained a wall

W of height $1 + R(f(k-1)+1, t, 3)$, then by iterating the lemma four times we could find vertices $x_1 \sim y_1 = x_2 \sim y_2 = x_3 \sim y_3 = x_4 \sim y_4 = x_5$ that induce P_5. This would be a contradiction.

Proof of Lemma 28. Fix an integer t, a graph $G \in \text{Forb}(P_5)$, and a vertex x such that $\omega(G) = k$ and G contains a wall $W = \{L_1, \ldots, L_{t'}\}$ of height $t' = 1 + R(f(k-1)+1, t, 3)$ with $x \in L_{t'}$. Then for every $s \in [t'-1]$, $N(x) \cap L_s \neq \emptyset$. Define a function g on the 2-subsets of [t-1] by

$$\begin{aligned}&\text{if } \forall v \in L_j \cap N(x) \exists u \in L_i \cap N(x) v \sim u, \text{ then } g(i<j) = 1\\ &\text{else if } \forall v \in L_j \cap N^c(x) \exists u \in L_i \cap N^c(x) v \sim u, \text{ then } g(i<j) = 2\\ &\text{else } g(i<j) = 3\ .\end{aligned}$$

Then by the choice of t', there exists a subset $S \subset [t'-1]$ such that all pairs in S have the same color α and either (1) $\alpha = 1$ and $|S| = f(k-1)+1$ or (2) $\alpha = 2$ and $|S| = t$ or (3) $\alpha = 3$ and $|S| = 3$. Note that if (1) holds, then $\{L_s \cap N(x) : s \in S\}$ is a wall, and if (2) holds, then $\{L_s \cap N^c(x) : s \in S\}$ is a wall.

Suppose (1) holds. Let F be the subgraph of G induced by $N(x)$. Since $\{L_s \cap N(x) : s \in S\}$ is a wall in F of height $f(k-1)+1$, the induction hypothesis implies that $\omega(F) \geq k$. Since x is adjacent to every vertex in F, $\omega(F) \geq k+1$, which is a contradiction. So (1) is impossible.

Suppose (3) holds. Say $S = \{q < r < s\}$. Since $g(q<r) \neq 1$ and $g(r<s) \neq 1$, there exist $v \in L_r \cap N(x)$ and $w \in L_s \cap N(x)$ such that v is not adjacent to any vertex in $L_q \cap N(x)$ and w is not adjacent to any vertex in $L_r \cap N(x)$. Thus $L_q \cap N^c(x) \cap N(v)$ and $L_r \cap N^c(x) \cap N(w)$ are nonempty. Let $u' \in L_q \cap N^c(x) \cap N(v)$. Since $1 \neq g(q<r) \neq 2$, there exists $v' \in L_r \cap N^c(x)$ such that v' is not adjacent to any vertex in $L_q \cap N^c(x)$. Thus v' is not adjacent to u' and there exists $u \in L_q \cap N(x) \cap N(v')$. Since v is not adjacent to any vertex in $L_q \cap N(x)$, v is not adjacent to u. Since L_q and L_r are independent sets, u is not adjacent to u' and v is not adjacent to v'. Thus (v', u, x, v, v') induces P_5, which is a contradiction. So (3) is impossible.

Thus (2) must hold. Let s be the largest element in S and let $y \in L_s \cap N(x)$. Let H be the subgraph of G induced by $\{y\} \cup \bigcup_{r \in S-\{s\}} \{L_r \cap N^c(x)\}$. Then y is the only vertex of H adjacent to x and $\{y\} \cup \{L_r \cap N^c(x) : r \in S - \{s\}\}$ is a wall in H of height t with top level $\{y\}$.

This completes the proofs of Lemma 28 and Theorem 27. □

Theorem 27 completely answers the question of whether Forb(T) is First-Fit χ-bounded for any tree T. Next consider the problem in which a tree and some other graphs are forbidden. The following theorem of this type is a key result in the proof of Theorem 24. Its proof requires a special Ramsey theoretic result due to Galvin, Rival, and Sands [6].

Theorem 29. (Kierstead and Penrice [21]).
For every tree T and complete bipartite graph $K_{t,t}$, Forb$(T, K_{t,t})$ is First-Fit χ-bounded.

Let B_t be the graph introduced in Example 6. Let D_k be the tree obtained by adding an edge between the roots of two disjoint copies of S_k, where S_k is the star on k vertices whose root is its only nonleaf.

Theorem 30. (Kierstead, Penrice and Trotter [23]).
The classes Forb$(P_{5,1}, B_t)$ and Forb(D_k, B_t) are First-Fit χ-bounded.

Consider a graph $G \in \text{Forb}(P_{5,1})$ on which First-Fit uses a huge number of colors. Theorem 30 explains why First-Fit uses at least a large numbers of colors on G. The reason is that G contains an induced subgraph that is either a large clique or a large induced B_t. Since First-Fit is known to require a large number of colors on either of these subgraphs, First-Fit requires a large number of colors on G. It would be very nice to prove a theorem of the following form: If $\chi_{\text{FF}}(G) > g(k)$, then G induces a graph in Q_k, where Q_k is a finite set of graphs such that $\chi_{\text{FF}}(H) \geq k$, for all $H \in Q_k$. The author, Penrice and Trotter [23] generalized the construction of B_t as follows. Call a graph $H = (V, E)$ *t-bad* if there exist sets $A_1, \ldots, A_t$ such that:

(1) $V = A_1 \cup \ldots \cup A_t$;
(2) $A_j = \{a_{1,j}, \ldots, a_{j,j}\}$ is an independent set of vertices for $j \in [t]$;
(3) $A_j \cap A_{j+1} = \emptyset$, for $j \in [t]$;
(4) $a_{i,k} \not\sim a_{i,j}$, whenever $1 \leq i \leq k < j \leq t$; and
(5) $a_{i,j} \sim a_{k,j+1}$, whenever $1 \leq i < k \leq j+1 \leq t$.

If H is t-bad then $\chi_{\text{FF}}(H) \geq t$. Note that it is not required that $A_k \cap A_j = \emptyset$, if $|j-k| \geq 2$. If $a_{i,j} = a_{i,j+2}$, whenever $1 \leq i \leq j \leq t-2$, then H is just B_t with one vertex removed. It is easy to see that it is possible to present the vertices so that $a_{i,j}$ precedes $a_{r,s}$ if $i < r$ and that when the vertices are presented in such an order, First-Fit uses t colors.

Problem 31. Let $Q_k = \{T_k\} \cup \{H : H \text{ is } k\text{-bad}\}$. Does there exist a function $g(k)$ such that if $\chi_{\text{FF}}(G) > g(k)$, then G contains an induced subgraph H such that $H \in Q_k$? □

We have already seen in Examples 5 and 6 that First-Fit performs optimally on trees. For interval orders, First-Fit is not optimal. Example 4 showed that there is an on-line algorithm that will color any interval graph G with $3\omega(G) - 2$ colors, while Chrobak and Slusarek showed the following.

Theorem 32. (Chrobak and Slusarek [3]).
There exists a constant C such that for every positive integer k, there exists an interval graph G with $\omega(G) = k$ such that First-Fit uses at least $4.4k - C$ colors on G. □

However the author answered a question of Woodall [37] by proving the following theorem that shows that First-Fit is close to optimal for interval graphs.

Theorem 33. (Kierstead [17]).
For any interval graph G, $\chi_{\text{FF}}(G) \leq 40\omega(G)$.

Proof. Identify the vertices of G with the intervals of an interval representation of G. Let $L \subset V$. An interval I has *density* $d = d(I/L)$ in L if every point in I is in at least d intervals in L. If K is a k-clique in G, then some interval in the representation of K has density at least $d/2$. To see this, alternately remove from K the intervals with the least left endpoint and the greatest right endpoint. When this process terminates the last interval will be contained in each pair of removed intervals and thus will have density at least $d/2$. Next we define the notion of *centrality*. Consider two adjacent intervals I and J in a set of intervals L. Let $N = N(J) \cap L$. Then N is a set of intervals and so defines an interval order P. Let $\lambda(I/J, L)$ be the length of the longest chain of intervals less than or equal to I in P. Note that if $I' \in N$ satisfies $\lambda(I/J, L) = \lambda(I'/J, L)$, then I is adjacent to I'. Similarly, let $\rho(I/J, L)$ be the length of the longest chain of intervals greater than or equal to I in P. Again, if $\rho(I/J, L) = \rho(I'/J, L)$, then I is adjacent to I'. The *centrality* of I in J with respect to L is $c(I/J, L) = \min\{\lambda(I/J, L), \rho(I/J, L)\}$. Note that $c(I/J, L) > 0$ iff $I \subset J$.

Let W be the wall of height h associated with a First-Fit coloring of G. We shall actually prove that W contains an interval with density $h/40$. Begin by setting I_1 equal to any interval in the top level of W and $d_1 = 1$. Now suppose that we have constructed a sequence $S = (I_1, \ldots, I_k)$ of intervals and a sequence $D = (d_1, \ldots, d_k)$ of integers. (I_1 and d_1 may have changed.) Let X_1 be the top $40d_1$ levels of W, X_2 be the next $40d_2$ levels of W, etc. Let T_t be the top $20d_t$ levels of X_t and B_t be the bottom $20d_t$ levels of X_t, for all $t \in [k]$. Also, let $B_{-1} = \phi$. We shall write X_t^*, T_t^*, and B_t^* for $\bigcup X_t$, $\bigcup T_t$, and $\bigcup B_t$, respectively. So for example X_t is a set of levels, while X_t^* is the set of intervals contained in the levels in X_t. Suppose further that the pair (S, D) is *acceptable*, i.e.,

(1) $c\left(I_{t+1}/I_t, B_t^* \cup T_{t+1}^*\right) \geq 2$, for all $t \in [k-1]$;
(2) $d\left(I_t/B_{t-1}^* \cup T_t^*\right) \geq d_t$, for all $t \in [k]$; and
(3) $d_{t+1} \geq 3^{-c+2}$, where $c = c\left(I_{t+1}/I_t, B_t^* \cup T_{t+1}^*\right)$, for all $t \in [k-1]$.

Then $d\left(I_k/X\right) \geq d = \sum D = \sum_{t \in [k]} d_t$, where X is the union of $\{X_1, \ldots, X_k\}$. It suffices to show that if $d < h/40$, then we can find a new acceptable pair (S', D') with $d' > d$, where $d' = \sum D'$. Note that in this case B_k is well defined and we have not yet used any of the intervals in B_k. We make further progress by taking advantage of these unused intervals.

Consider $N = N\left(I_k\right) \cap B_k^*$. As above N can be partitioned into cliques $K_0, K_0', \ldots, K_c, K_c'$ so that for all indices i, $\lambda\left(I/I_k, B_t^*\right) = i = c\left(I/I_k, B_t^*\right)$, for all intervals $I \in K_i$ and $\rho\left(I/I_k, B_t^*\right) = i = c\left(I/I_k, B_t^*\right)$, for all intervals $I \in K_i'$. Since $|N| \geq 20d_k = 2(1+2+2+5)$, either

(i) $|K_i| \geq 2\left\lceil 3^{-i+2} d_k\right\rceil$ or $|K_i'| \geq 2\left\lceil 3^{-i+2} d_k\right\rceil$, for some i such that $3 \leq i \leq \log_3 d_k$,
(ii) $|K_2| \geq 2d_k$, $|K_2'| \geq 2d_k$, $|K_1| \geq 2d_k$, or $|K_1'| \geq 2d_k$,
(iii) $|K_0| \geq 5d_k$ or $|K_{0'}| \geq 5d_k$.

(We can improve the argument at this point by showing that $|N|$ is actually considerably larger than $20d_k$.) In each case, we assume without loss of generality that the condition is met by K_i.

Case (i). The clique K_i contains an interval I_{k+1} such that $d(I_{k+1}/B_k^*) \geq 3^{-i+2}d_k$ and $c(I/I_k, B_t^*) = i \geq 2$. Thus setting $S' = (I_1, \ldots, I_k, I_{k+1})$ and $D' = (d_1, \ldots, d_k, \lceil 3^{-i+2}d_k \rceil)$ yields an acceptable pair (S', D') with $d' > d$.

Case (ii). Let $|K_i| \geq 2d_k$, where $i \in \{1, 2\}$. Then K_i contains an interval I such that $d(I/B_k^*) \geq d_k$ and $c(I/I_k, B_k^*) \geq 1$. Thus $I \subset I_k$. It follows that $d(I/B_{k-1}^* \cup T_k^* \cup B_k^*) \geq 2d_k$ and $c(I/I_{k-1}, B_{k-1}^* \cup T_k^* \cup B_k^*) \geq c(I_k/I_{k-1}, B_{k-1}^* \cup T_k^* \cup B_k^*) \geq 2$. Note that $T_k \cup B_k$ consists of the next $21^*(2d_k)$ levels after B_{k-1}. Thus setting $S' = (I_1, \ldots, I_{k-1}, I)$ and $D' = (d_1, \ldots, d_{k-1}, 2d_k)$ yields an acceptable pair (S', D') with $d' > d$.

Case (iii). The left endpoint of d_k is contained in $5d_k$ intervals from $K_0 \subset B_k^*$ and d_k additional intervals from $B_{k-1}^* \cup T_k^*$ that witness that $d(I_k/B_{k-1}^* \cup T_k^*) \geq d_t$. Thus for one of these intervals I, $d(I/B_{k-1}^* \cup T_k^* \cup B_k^*) \geq 3d_k$. Let J be the interval in $B_{k-1}^* \cup T_k^*$ with $c(J/I_{k-1}, B_{k-1}^* \cup T_k^*) = c-1$, where $c = c(I_k/I_{k-1}, B_{k-1}^* \cup T_k^*)$. If $J \subset I$, let $I' = J$; otherwise let $I' = I$. In either case $c(I'/I_{k-1}, B_{k-1}^* \cup T_k^* \cup B_k^*) \geq c-1$ and $d(I'/B_{k-1}^* \cup T_k^* \cup B_k^*) \geq 3d_k$. First suppose that $c > 2$. Note that $T_k \cup B_k$ is contained in the first 21^*3d_k levels below B_{k-1}. Thus setting $S' = (I_1, \ldots I_{k-1}, I')$ and $D' = (d_1, \ldots, d_{k-1}, 3d_k)$ yields an acceptable pair (S', D') with $d' > d$. Otherwise $c = 2$. Then $d_k \geq d_{k-1}$ and $I' \subset I_{k-1}$. Thus $d(I'/B_{k-2}^* \cup T_{k-1}^* \cup B_{k-1} \cup T_k^* \cup B_k^*) \geq 2d_{k-1} + 2d_k$ and $c(I'/I_{k-2}, B_{k-2}^* \cup T_{k-1}^* \cup B_{k-1}^* \cup T_k^* \cup B_k^*) \geq c(I_{k-1}/I_{k-2}, B_{k-2}^* \cup T_{k-1}^* \cup B_{k-1}^* \cup T_k^* \cup B_k^*)$. Thus setting $S' = (I_1, \ldots, I_{k-2}, I')$ and $D' = (d_1, \ldots, d_{k-2}, 2d_{k-1} + 2d_k)$ yields an acceptable pair (S', D') with $d' > d$. □

The proof of Theorem 4.7 can be strengthened to obtain a better constant. The author and Qin [24] showed that at most $26\omega(G)$ colors are used by First-Fit to color interval graphs. I strongly believe that the real constant is less than ten.

A graph $G = (V, E)$ is *d-degenerate* (sometimes called d-inductive) if there exists an ordering $v_1 \prec v_2 \prec \cdots \prec v_n$ of the vertices of G so that $|\{w \in V : w \sim v \text{ and } w \prec v\}| \leq d$, for all $v \in V$. For example planar graphs are 5-inductive and trees are 1-inductive. If G is a d-degenerate graph, then clearly $\chi(G) \leq d + 1$. The graph G is chordal if every cycle in G of length at least four contains a chord, i.e, an edge between two nonconsecutive vertices. There are two well known characterizations of chordal graphs. The first is that G is chordal iff there exists an ordering $v_1 \prec v_2 \prec \cdots \prec v_n$ of the vertices of G so that $\{w \in V : w \sim v \text{ and } w \prec v\}$ is a clique, for all $v \in V$. Thus if G is chordal, then G is $(\omega(G) - 1)$-degenerate. It follows that chordal graphs are perfect. The other characterization is that G is chordal iff there exists a mapping f of the vertices of G to subtrees of a tree T such that two vertices u and v are adjacent iff $E(f(u)) \cap E(f(v)) \neq \emptyset$. Thus interval graphs are chordal since intervals can be represented as subpaths of a path. Irani showed that First-Fit performs close to optimally on the class of d-degenerate graphs.

Theorem 34. (Irani [12]).
There exists a constant C such that First-Fit uses at most $Cd \log n$ colors to

color any d-degenerate graphs on n vertices. Moreover there exists a constant C' such that for every on-line algorithm A and all integers d and n with $n > d^3$, there exists a d-degenerate graph G on n vertices such that $\chi_A(G) \geq C'd\log n$.

We end this section with two more applied problems. First we consider the Broadcast Problem, which is a generalization of Path Coloring on Trees.

Broadcast Problem.
INSTANCE: Tree T and a set $S = \{T_1, \ldots, T_n\}$ of subtrees of T.
PROBLEM: Color the elements of S with as few colors as possible subject to the condition that two subtrees that share an edge must receive different colors.

By the discussion above, the Broadcast Problem is equivalent to coloring chordal graphs. Thus by Theorem 34 First-Fit uses at most $\chi \log n$ colors, where χ is the optimal number of colors. If the underlying tree of the Broadcast problem is a path, then the problem reduces to interval graph coloring and First-Fit uses at most 42χ colors. Irani left open the problem of whether there exists a chordal graph G on n vertices for which First-Fit uses $\Omega(\omega(G)\log n)$ colors.

The second application is the following storage problem that was shown to be NP-complete by Stockmeyer [32]. Here we shall only be interested in the off-line version; however we will use on-line interval graph coloring to obtain a polynomial time approximation for this off-line version.

Dynamic Storage Allocation (DSA).
INSTANCE: Set A of items to be stored, each $a \in A$ having a positive integer size $s(a)$, a non-negative integer arrival time $r(a)$, and a positive integer departure time $d(a)$, and a positive storage size D.
PROBLEM: Is there a feasible allocation of storage for A, i.e., a function $\sigma : A \to \{1, 2, \ldots, D\}$ such that for every $a \in A$ the allocated storage interval $I(a) = [\sigma(a), \sigma(a)+1, \ldots, \sigma(a)+s(a)-1]$ is contained in $[1, D]$ and such that, for all $a, a' \in A$, if $I(a) \cap I(a')$ is nonempty then either $d(a) \leq r(a')$ or $d(a') \leq r(a)$?

Notice that if all the items have the same size, then DSA is just interval graph coloring. The following approximation algorithm for DSA has been well known since the late sixties. First put each item into the smallest possible box whose size is a power of two. Then order the boxes by decreasing size and use First-Fit to assign the boxes to storage locations. Clearly this algorithm produces a proper assignment of storage locations. Let ω^* be the maximum, over all times t, of the sum of the sizes of the boxes that must be stored at time t. Then ω^* is a lower bound on the amount of storage that must be used. It is easy to show that, because of the uniformity in box size and the use of First-Fit, this algorithm uses at most $f(\omega^*)$ storage locations, where $f(k)$ is the maximum number of colors used by First-Fit to color an interval graph with clique number k. Thus Theorem 34 has the following corollary.

Corollary 35. (Kierstead [17]).
There is a polynomial time approximation algorithm for Dynamic Storage Allocation with a constant performance ratio of 80. □

It later turned out that a slight modification of the algorithm in Example 4 had this same property. So we have the following corollary to Example 4.

Corollary 36. (Kierstead [18]).
There is a polynomial time approximation algorithm for Dynamic Storage Allocation with a constant performance ratio of six. □

This is an example of the following hypothetical situation. An off-line optimization problem (DSA) can be reduced to a simpler problem (interval graph coloring) if the data is preordered in a certain way (decreasing box size). However this preordering turns the simpler problem into an on-line problem. I know of no other examples of this situation, but if it turns out that there are other examples, this could be a very important application for on-line theory.

Acknowledgements

This work was partially supported by Office of Naval Research grant N00014-90-J-1206.

References

1. D. Bean. Effective coloration. *J. Symbolic Logic*, 41:469–480, 1976.
2. A. Blum and D. Karger. An $\tilde{O}(n^{3/14})$-coloring for 3-colorable graphs. Preprint.
3. M. Chrobak and M. Ślusarek. On some packing problems relating to Dynamical Storage Allocations. *RAIRO Informatique Theoretique*, 22:487–499, 1988.
4. P. Erdős and A. Hajnal. On chromatic numbers of graphs and set systems. *Acta Math. Sci. Hung.*, 17:61–99, 1966.
5. S. Felsner. On-line chain partitions of orders. To appear in *Theoret. Comput. Sci.*
6. F. Galvin, I. Rival, and B. Sands. A Ramsey-type theorem for traceable graphs. *J. Comb. Theory, Series B*, 33:7–16, 1982.
7. A. Gyárfás. On Ramsey covering numbers. In *Coll. Math. Soc. János Bolyai 10, Infinite and Finite Sets*, pages 801–816. North-Holland/American Elsevier, New York, 1975.
8. A. Gyárfás. Problems from the world surrounding perfect graphs. *Zastowania Matematyki Applicationes Mathemacticae*, 19:413–441, 1985.
9. A. Gyárfás and J. Lehel. On-line and first-fit colorings of graphs. *Journal of Graph Theory*, 12:217–227, 1988.
10. A. Gyárfás and J. Lehel. Effective on-line coloring of P_5-free graphs. *Combinatorica*, 11:181–184, 1991.
11. A. Gyárfás, E. Szemerédi, and Z. Tuza. Induced subtrees in graphs of large chromatic number. *Discrete Math*, 30:235–244, 1980.
12. S. Irani. Coloring inductive graphs on-line. *Algorithmica*, 11:53–72, 1994. Also in Proc. 31st IEEE Symposium on Foundations of Computer Science, 1990, 470-479.
13. H. A. Kierstead. On-line coloring k-colorable graphs. To appear in *Israel J. of Math.*
14. H. A. Kierstead. An effective version of Dilworth's theorem. *Trans. Amer. Math. Soc.*, 268:63–77, 1981.

15. H. A. Kierstead. Recursive colorings of highly recursive graphs. *Canadian J. Math.*, 33:1279–1290, 1981.
16. H. A. Kierstead. An effective version of Hall's theorem. *Proc. Amer. Math. Soc.*, 88:124–128, 1983.
17. H. A. Kierstead. The linearity of first-fit coloring of interval graphs. *SIAM Journal on Discrete Math*, 1:526–530, 1988.
18. H. A. Kierstead. A polynomial time approximation algorithm for dynamic storage allocation. *Discrete Math.*, 88:231–237, 1991.
19. H. A. Kierstead and K. Kolossa. On-line coloring of perfect graphs. To appear in *Combinatorica.*
20. H. A. Kierstead and S. G. Penrice. Recent results on a conjecture of Gyárfás. *Congressus Numerantium*, 79:182–186, 1990.
21. H. A. Kierstead and S. G. Penrice. Radius two trees specify χ-bounded classes. *J. Graph Theory*, 18:119–129, 1994.
22. H. A. Kierstead, S. G. Penrice, and W. T. Trotter. On-line graph coloring and recursive graph theory. *Siam Journal on Discrete Math.*, 7:72–89, 1994.
23. H. A. Kierstead, S. G. Penrice, and W. T. Trotter. First-fit and on-line coloring of graphs which do not induce P_5. *Siam Journal on Discrete Math.*, 8:485–498, 1995.
24. H. A. Kierstead and J. Qin. Coloring interval graphs with First-Fit. *Discrete Math.*, 144:47–57, 1995.
25. H. A. Kierstead and W. T. Trotter. An extremal problem in recursive combinatorics. *Congressus Numerantium*, 33:143–153, 1981.
26. L. Lovász. *Combinatorial Problems and Exercises.* Akadémiai Kiadó, Budapest, 1993. Problem 13.13.
27. L. Lovász, M. Saks, and W. T. Trotter. An on-line graph coloring algorithm with sublinear performance ratio. *Discrete Mathematics*, 75:319–325, 1989.
28. A. B. Manaster and J. G. Rosenstein. Effective matchmaking (recursion theoretic aspects of a theorem of Philip Hall). *Proc. Lond. Math. Soc.*, 25:615–654, 1972.
29. S. G. Penrice. On-line algorithms for ordered sets and comparability graphs. Preprint.
30. J. H. Schmerl. Private communication (1978).
31. J. H. Schmerl. Recursive colorings of graphs. *Canad. J. Math.*, 32:821–830, 1980.
32. I. J. Stockmeyer. Private communication (1976).
33. D. P. Sumner. Subtrees of a graph and chromatic number. In Gary Chartrand, editor, *The Theory and Applications of Graphs*, pages 557–576. John Wiley & Sons, New York, 1981.
34. M. Szegedy. Private communication (1986).
35. E. Szemerédi. Private communication (1981).
36. S. Vishwanathan. Randomized on-line graph coloring. *J. Algorithms*, 13:657–669, 1992. Also in Proc. 31st IEEE Symposium on Foundations of Computer Science, 1990, 464–469.
37. D. R. Woodall. Problem no. 4, combinatorics. In T. P. McDonough and V. C. Marvon, editors, *London Math. Soc. Lecture Note Series 13.* Cambridge University Press, 1974. Proc. British Combinatorial Conference 1973.