COMBINATORICA
Akadémiai Kiadó – Springer-Verlag

COMBINATORICA 16 (4) (1996) 479–491

ON-LINE COLORING OF PERFECT GRAPHS

H. A. KIERSTEAD[1] and K. KOLOSSA

Received November 10, 1994

Lovász, Saks, and Trotter showed that there exists an on-line algorithm which will color any on-line k-colorable graph on n vertices with $O(n\log^{(2k-3)} n/\log^{(2k-4)} n)$ colors. Vishwanathan showed that at least $\Omega(\log^{k-1} n/k^k)$ colors are needed. While these remain the best known bounds, they give a distressingly weak approximation of the number of colors required. In this article we study the case of perfect graphs. We prove that there exists an on-line algorithm which will color any on-line k-colorable perfect graph on n vertices with $n^{10k/\log\log n}$ colors and that Vishwanathan's techniques can be slightly modified to show that his lower bound also holds for perfect graphs. This suggests that Vishwanathan's lower bound is far from tight in the general case.

0. Introduction

An *on-line graph* is a structure $G^< = (V,E,<)$, where $G=(V,E)$ is a graph, and $<$ is a linear ordering of V. We call $<$ an *on-line presentation* of the graph G. The on-line subgraph of $G^<$ induced by a subset $X \subset V$ is the on-line graph $G^<[X]=(X,E',<')$, where E' is the set of edges in E both of whose end points are in X and $<'$ is $<$ restricted to X. Let $V_i=\{x_1,\dots,x_i\}$ denote the first i vertices of V in the linear order $<$ and set $G_i^< = G^<[V_i]$. An algorithm for coloring the vertices of an on-line graph $G^<$ is said to be *on-line* if the color of a vertex x_i is determined solely by $G_i^<$. Intuitively, the algorithm receives the vertices of $G^<$ one at a time in some externally determined order $x_1,\dots,x_n$. At the stage i when the vertex x_i is received, the algorithm can only see $G_i^<$. At this stage, based only on this current information, the algorithm must irrevocably assign a color to the vertex x_i. The algorithm First-Fit is a simple, but important example of an on-line algorithm. First-Fit colors the vertices of G with an initial sequence of the colors $\{1,2,\dots\}$ by assigning to the vertex v_i the least color not already assigned to some vertex $x\in V_{i-1}$ such that x is adjacent to v_i.

Let $G^< = (V,E,<)$ be an on-line graph and $\mathbf{A}$ be an on-line algorithm. The number of colors that $\mathbf{A}$ uses to color $G^<$ is denoted by $\chi_{\mathbf{A}}(G^<)$. For a class of graphs Γ, let $\chi_{\mathbf{A}}(\Gamma)$ denote the maximum value of $\chi_{\mathbf{A}}(G^<)$ over all $G\in\Gamma$ and all

Mathematics Subject Classification (1991): 05 C

[1] Research partially supported by Office of Naval Research grant N00014-90-J-1206.

on-line presentations $<$ of G. The *on-line chromatic number*, denoted by $\chi_{\mathrm{ol}}(\Gamma)$, of the class Γ is the minimum of $\chi_{\mathbf{A}}(\Gamma)$ over all on-line algorithms $\mathbf{A}$. We say that Γ is *on-line χ-bounded* if there is a function f and an on-line algorithm $\mathbf{A}$ such that $\chi_{\mathbf{A}}(G) \leq f(\omega(G))$, for every $G \in \Gamma$, where $\omega(G)$ denotes the clique number of G. In this case f is said to be a *χ-binding function* for the class Γ. Let $\Gamma(k,n)$ be the class of k-colorable graphs on n vertices.

A considerable amount of effort has been put into finding classes of graphs that are on-line χ-bounded. The largest collection of such classes was obtained by Kierstead, Penrice, and Trotter [7]. For a fixed tree T with radius at most two, they showed that the class Forb(T), consisting of all graphs that do not induce T, is on-line χ-bounded. Many results are known about on-line coloring of various classes of perfect graphs. The result in [6] shows that the class of co-comparability graphs is on-line χ-bounded. Kierstead and Trotter [8] showed that the on-line chromatic number of the class of interval graphs is $3\omega-2$. Gyárfás and Lehel [2] showed that, even using First-Fit, the classes of split graphs, complements of bipartite graphs, and complements of chordal graphs have linear χ-binding functions. Kierstead [5] proved that First- Fit only requires a linear number of colors on the class of interval graphs.

Bean [1] showed that the class of trees is not on-line χ-bounded. Thus the superclasses of comparability graphs, chordal graphs, and perfect graphs are not on-line χ-bounded. Lovász, Saks, and Trotter [9] observed that the on-line chromatic number of the class $\Gamma(2,n)$ (which includes trees) is $\Theta(\log n)$. This result leads naturally to the problem of bounding $\chi_{\mathrm{ol}}(\Gamma(k,n))$ in terms of k and n. Of course $\chi_{\mathrm{ol}}(\Gamma(k,n)) \leq n$, since we can color every vertex with a different color. As with ordinary graph coloring and polynomial time graph coloring, a big jump in complexity occurs between $\Gamma(2,n)$ and $\Gamma(3,n)$. With considerable effort, Lovász et al improved the trivial upper bound by proving the following (barely) sublinear bound, where $\log^{(k)}$ is the kth iteration of the logarithm function.

$$\chi_{\mathrm{ol}}(\Gamma(k,n)) = O(n \log^{(2k-3)} n / \log^{(2k-4)} n).$$

When $n \leq k2^k$, Szegedy [10] proved that this bound is close to optimal. He showed that then

$$\chi_{\mathrm{ol}}(\Gamma(k,n)) > n/k.$$

For n larger than $k2^k$ Szegedy's technique breaks down. Vishwanathan showed that

$$\chi_{\mathrm{ol}}(\Gamma(k,n)) = \Omega(\log^{k-1} n / k^k).$$

While these remain the best known bounds, they provide a distressingly weak approximation of $\chi_{\mathrm{ol}}(\Gamma(k,n)$ for fixed k. Irani [4] showed that much more is possible for chordal graphs. She proved that for the class $C(k,n)$ of k-colorable chordal graphs on n vertices

$$\chi_{\mathrm{ol}}(C(k,n)) = \Theta(k \log n).$$

In fact she showed that this performance is achieved by the on-line algorithm First-Fit. However this is not true for all perfect graphs. It is known that there are

2-colorable (and thus perfect) graphs on n vertices for which First-Fit requires $n/2$ colors. In this article we shall consider the class of perfect graphs. Our main result is the following theorem.

Theorem 0.1. *Let k be a positive integer. There exists an on-line algorithm $\mathbf{A}_k$ such that $\mathbf{A}_k$ colors every on-line k-colorable perfect graph $G^<$ on n vertices with at most $n^{10/\log\log n}$ colors.*

We also show that Vishwanathan's construction can be slightly modified to produce perfect graphs. This result, which is stated below, leads us to suspect that we have not yet been clever enough in constructing on-line graphs to witness lower bounds in the general case.

Theorem 0.2. *For every fixed positive integer k and every on-line algorithm $\mathbf{A}$, there exists an on-line k-colorable perfect graph $G^<$ such that $\mathbf{A}$ uses at least $\frac{1}{2}\left(\frac{\log_3 n}{2k}\right)^{k-1}$ colors on $G^<$.*

We shall use the following notation with respect to a fixed graph $G=(V,E)$. Adjacency between vertices x and y is denoted by $x \sim y$. For vertices x and y, the *distance* $d(x,y)$ from x to y is the number of edges in the shortest path from x to y. Let $X \subset V$. For a non-negative integer r, the *closed r-neighborhood* $N^r[X]$ of X is the set $\{v \in V : d(x,v) \leq r, \quad \text{for some} \quad x \in X\}$. For a positive integer r the *open r-neighborhood* $N^r(X)$ of X is the set $N^r(X) = N^r[X] - N^{r-1}[X]$. We write $N(X)$ and $N[X]$ for $N^1(X)$ and $N^1[X]$. We also slightly abuse notation by writing $N^r(x)$ and $N^r[x]$ for $N^r(\{x\})$ and $N^r[\{x\}]$. The clique number of G is denoted by $\omega(G)$. The set $\{1,\dots,n\}$ is denoted by $[n]$. The function with domain $\emptyset$ is denoted by ϵ.

1. Preliminaries

The following lemma provides the key property of perfect graphs that we exploit to obtain improved upper bounds on the on-line chromatic number of this class.

Lemma 1.1. *Let v be a vertex in a perfect graph G. Suppose that I_0, I_1, … I_r, I_{r+1} are subsets of V such that*

(1) *$I_0 = \{v\}$;*

(2) *I_j is an independent set, for all $j \leq r$;*

(3) *$I_{j+1} \subset N(I_j)$, for all $j \leq r$; and*

(4) *If $x \sim y$ with $x \in l_i$ and $y \in I_j$, then $|i-j|=1$.*

Then $\omega(I_{r+1}) < \omega(G)$.

Proof. We first claim that if x and y are adjacent vertices in I_{r+1}, then $N(x) \cap I_r \subset N(y) \cap I_r$ or vice versa. Otherwise there exists $p_r \in I_r$ such that $p_r \sim x$, but not

$p_r \sim y$ and there exists $q_r \in I_r$ such that $q_r \sim y$, but not $q_r \sim x$. Let $P = \{p_0, p_1, \dots p_r\}$ and $Q = \{q_0, q_1, \dots, q_r\}$ be paths such that $p_j,\ q_j \in I_j$. Let j be the largest index such that $p_j \sim q_{j+1}$ or $p_{j+1} \sim q_j$. Note that j exists since $p_0 = v = q_0$. Say $p_j \sim q_{j+1}$. Then $p_j, \dots p_r, x, y, q_r, \dots q_{j+1}$ is an induced odd cycle of length at least five, which is a contradiction.

Now suppose that $K \subset I_{r+1}$ is a clique. Then by the above claim there exists a vertex $x \in K$ such that for all $y \in K$, $\emptyset \neq N(x) \cap I_r \subset N(y) \cap I_r$. Thus there exists $k \in I_r$ such that $K \cup \{k\}$ is a larger clique than K. ∎

Corollary 1.2. *Let $G^<$ be an on-line graph such that G is perfect and k-colorable. Then there exists an on-line algorithm which, for a specified integer t, a subset $S \subset V_t$ with $G[S]$ connected, and a vertex $v \in S$, partitions $N(S) \cap (V - V_t)$ into fewer than k^r parts, where r is the maximum distance in $G[S]$ of any vertex in S to v, such that each part induces a $(k-1)$-colorable subgraph.*

Proof. The algorithm begins at vertex v_{t+1}, at which time $G[S]$ has already been presented. Thus we may fix a proper k-coloring c of $G[S]$. We will partition $N(S) \cap (V - V_t)$ into parts F_σ, where each part is indexed by a sequence $\sigma = (\sigma_1, \dots \sigma_s)$ of colors, where $s \leq r$. When presented with $z \in N(S) \cap (V - V_t)$, choose an arbitrary shortest path $P_z = v, v_1, \dots, v_s, z$ from v to z in $G[S \cup \{z\}]$ and assign z to the part F_σ, with $\sigma = (\sigma_1, \dots \sigma_s)$, where σ_i is the color of v_i.

To see that F_σ has clique size less than k (and hence, by perfection of G, is $(k-1)$-colorable), we apply Lemma 1.1 as follows. For a given $\sigma = (\sigma_1, \dots \sigma_s)$, let V_σ be the union of the $P_z - \{z\}$ such that z is assigned to part F_σ. For $j \leq s+1$, define I_j to be the subset of V_σ whose distance from v in $G[V_\sigma]$ is j. Then $I_0, \dots, I_{s+1}$ satisfy the hypotheses of Lemma 1.1, and I_{S+1} is the set of elements of $N(S) \cap (V - V_t)$ that are assigned to F_σ. By Lemma 1.1, each F_σ induces a subgraph of G with clique size less than k. Since c is a proper coloring, for all non-empty F_σ, both $\sigma_{(1)} \neq c(v)$ and for all i, $\sigma_{(i)} \neq \sigma_{(i+1)}$. Thus $|F| \leq \sum_{i=0}^{r} (k-1)^i \leq \frac{(k-1)^{r+1} - 1}{k-2} \leq k^r$. ∎

2. The main result

We begin by proving a weaker version of Theorem 1 in which the algorithm knows ahead of time the number n of vertices that the input graph G has. We then use this weaker result to obtain the full theorem.

Theorem 2.1. *Let k and n be fixed positive integers. Then there exists an on-line algorithm $\mathbf{A}_{k,n}$ such that $\mathbf{A}_{k,n}$ colors every on-line k-colorable perfect graph $G^<$ on n vertices with at most $\frac{1}{4} n^{10k/\log\log n}$ colors.*

Proof. We argue by induction on k. The base step $k = 1$ is trivial and the case $k = 2$ follows from the remark of Lovász et al mentioned in the introduction. For the induction step $k \geq 3$, assume that for all j with $1 \leq j < k$ we have obtained an on-line $\mathbf{A}_{j,n}$ such that $\mathbf{A}_{j,n}$ uses $a(j,n) \le \frac{1}{4} n^{10j/\log\log n}$ colors to color any on-line j-colorable perfect graph. We must specify an on-line algorithm $\mathbf{A}_{k,n}$ such that $\mathbf{A}_{k,n}$ uses $a(k,n) \le \frac{1}{4} n^{10k/\log\log n}$ colors to color any on-line k-colorable perfect graph. We may assume that $10k - \log 4 \le \log\log n$, since otherwise we are allowing the algorithm to use n colors. Since $k \ge 3$, $28 \le \log\log n$. Thus we have the following inequality that will be used later.

$$(0) \qquad \log\log n \log\log\log n < \log^{l/2} n.$$

Let $f(j) = n^{4j/\log\log n}$, $R(j) = \lfloor \frac{5}{2} \cdot 6^j \rfloor - 2$, and $r(j) = 6^j - 1$. We will describe the algorithm in stages, where a stage is the period of calculation between the time that a new vertex x is presented and the time that the next vertex is presented. During the stage that a new vertex x is presented it will be permanently identified as a root or a nonroot. At this time, if x is a root, then x will have a priority $p(x)$, a rank $\varrho(x)$, a sphere (of influence) $S(x)$, a sphere coloring c_x, and a witness set $W(x)$. At various stages in the algorithm the priority of a root may increase, but it will never decrease. At these stages, and only these stages, the rank, sphere, sphere coloring, and witness set of the root will be reassigned. If x is a non-root, x will be permanently assigned a root v and a priority $p(x)$.

We begin with an overview of the steps performed during the stage that x is presented.

- S1. Update the priority, witness set, sphere, and sphere coloring of each previously presented root.
- S2. Update the rank of each previously presented root.
- S3. Decide whether x is a root or a nonroot and make the necessary assignments.
- S4. Color x, if x is a root.
- S5. Color x, if x is a nonroot.

Next we specify the computations that are made at each of the above steps.

Step S1: The previously presented roots are considered in the order of their rank with the highest ranking roots first. When a root v is considered its priority is changed to

$$p = \max\{j : |N^{r(j)}[v] \setminus \cup\{W(y) : y \neq v \quad \text{and} \quad p(y) \ge j\}| \ge f(j)\},$$

if p is greater than its current priority $p(v)$; otherwise its priority remains the same. If the priority $p(v)$ is changed, then the sphere and witness set of v are changed to

$$S(v) = N^{R(p(v))}[v] \quad \text{and}$$
$$W(v) = N^{r(p(v))}[v] \setminus \cup\{W(y) : y \neq v \quad \text{and} \quad p(y) \ge p(v)\}$$

and the sphere coloring c_v of v is changed to any proper k-coloring of the sphere $S(v)$ of v. (Note that this is not an on-line coloring; the sphere coloring is chosen only after the entire sphere is presented. Whenever the priority, and thus the sphere, of a root is changed a completely new sphere coloring is chosen.) ∎

Notice that while different roots can have overlapping spheres and witness sets, any two roots with the same priority j will have disjoint witness sets with cardinality at least $f(j)$. Thus there are at most $n/f(j)$ roots with priority j. Since $f(j)=n^{4j/\log\log n}>n$ when $j>\frac{1}{4}\log\log n$, we have:

$$(1) \qquad p(v) \leq \frac{1}{4}\log\log n < \frac{1}{4}n^{1/\log\log n}, \quad \text{for every root} \quad v.$$

Step S2: The updated ranks are calculated by the following rule:
$\varrho(v)<\varrho(x)$ iff $p(v)<p(x)$ or both $p(v)=p(x)$ and x is presented before v. ∎

Step S3: The vertex x is declared to be a nonroot if it is in the neighborhood of the current sphere of some root; otherwise it is declared to be a root. If x is a nonroot, then the root of x is the highest ranking root v such that $x\in N(S(v))$. The priority $p(x)$ of x is set equal to the current priority of v. (The root v of x and the priority $p(x)$ of x will never change even if the priority of v changes.) If x is a root, the priority $p(x)$ of x is set equal to 0 and the witness set, sphere, and sphere coloring are calculated as in Step S1. ∎

Step S4: If x is a root, color x with color 1. ∎

Since x is a root, it cannot be in the neighborhood of any sphere. Since each root is a member of its own sphere, x is not adjacent to any other root. Thus the set of roots is independent and so all can be colored with color 1.

Step S5: If x is a nonroot, we shall assign x a three coordinate color $(p(x),l(x),g(x))$. The first coordinate assures that two vertices with different priorities have different colors. The second coordinate, called the local color, will be calculated in Step 5i. It will assure that two vertices with the same root and priority get different colors. The third coordinate, called the global color, will be calculated in Step 5ii. It will assure that two vertices with the same priority but different roots get different colors.

Let $L(j)$ and $G(j)$ be the maximum number of local and global colors used on vertices of priority j. Then we have the following upper bound on the total number of colors used.

$$(2) \qquad a(k,n) \leq 1 + \sum_{j=0}^{\frac{1}{4}\log\log n} L(j)G(j).$$

Step S5i: The local color $l(x)=(\sigma_x,d_x)$ is itself a two coordinate color. To calculate the first coordinate, let v be the root of x. By Corollary 1.2, there is an on-line algorithm that partitions the neighbors of $S(V)$ presented after v obtains priority $p(x)$ into parts V_i, $i\in[k^{R(p(x))}]$ such that each $G[V_i]$ has clique size less than k. Let $\sigma_x=i$, where $x\in V_i$. Let d_x be the color assigned to x by $A_{k-1,n}$ when applied to the on-line graph $G^{<}[V_{\sigma_x}]$. ∎

Clearly this gives a proper coloring of all vertices with fixed priority j and fixed root v. Note that

$$L(j) \leq k^{R(j)} a(k-1,n).$$

Since $a(k-1,n) \leq n^{10(k-1)/\log\log n}$ and, using (0),

$$k^{R(j)} \leq k^{\frac{5}{2}\cdot 6^{\frac{\log\log n}{4}}} \leq e^{e^{\frac{\log\log n}{2}}\log k} \leq n^{\frac{\log\log\log n}{\log^{1/2} n}} \leq n^{1/\log\log n}, \tag{3}$$

we have

$$L(j) \leq n^{(10k-9)/\log\log n}. \tag{4}$$

Step 5ii: The global color $g(x)$ is obtained from an auxiliary digraph $A_j = (R_j, D_j)$, using a technique from [6]. Let R_j be the set of all roots that at some stage have priority $j = p(x)$. For distinct v and w in R_j, $V \to W$ is in D_j at stage s iff at stage s there exist y with root v and priority j and z with root w and priority j such that $y \sim z$ and z was presented before y. Notice that at a given stage $v \to w$ might not be in D_j, but at a later stage $v \to w$ might enter D_j. Once $v \to w$ enters D_j, it will remain in D_j. Thus we cannot hope to on-line color A_j. However we shall see that by changing the colors of the vertices relatively infrequently, we can maintain a proper coloring of A_j. Also note for future reference that if $v \to w$ is in D_j, then $d(v,w) \leq 2R(j)+3$.

Let the maximum outdegree of A_j be Δ_j^+ and the current out degree of a root v be $d^+(v)$. We shall maintain a coloring γ_j of A_j such that:

(i) At the end of any stage, if $v \to w$ in A_j, then the current color of v was never the color of w;

(ii) the color of a root changes only when its outdegree changes; and

(iii) at most $(\Delta_j^+ + 1)^2$ colors are used.

To do this, at each stage we assign the node r a two coordinate color $\gamma_j(r) = (d^+(r), t(r))$. If $d^+(r)$ is unchanged from the previous stage, i.e., we have not discovered any new outneighbors of r in A_j, then $\gamma_j(r)$ remains unchanged. If $d(r)$ is changed, then r is the only vertex whose out degree has changed. Then we choose $t(r) \in [d^+(r)+1]$ so that $(d^+(r), t(r))$ was never the color of any of the $d^+(r)$ outneighbors of r in A_j. This is possible since, using (ii), for each outneighbor q of r there is at most one second coordinate t such that at some stage q had color $(d^+(r), t)$. Clearly the conditions (i)–(iii) are all satisfied.

Let $g(x)$ be the current value of $\gamma_j(v)$, where v is the root of x. Note that $g(x)$ will never change, even if $\gamma_j(v)$ does change. ∎

We claim that if a previously presented nonroot y is adjacent to x, has the same priority j as x, but has a different root w than the root v of x, then y receives a different global color than x. Note that at the current stage (when x is presented)

$v \dashrightarrow w$ in A_j and $g(a) = \gamma_j(v)$. By (i), $\gamma_j(w)$ was never equal to the current value $g(x)$ of $\gamma_j(v)$. On the other hand, at the stage that y was presented, $g(y)$ was set equal to $\gamma_j(w)$. Thus $g(y) \neq g(x)$.

This completes the description of the on-line algorithm $\mathbf{A}_{k,n}$. By our remarks above, $\mathbf{A}_{k,n}$ gives a proper coloring of any on-line k-colorable perfect graph. It remains to show that $\mathbf{A}_{k,n}$ requires only $\frac{1}{4}n^{10k/\log\log n}$ colors to color any on-line k-colorable perfect graph. For this purpose we will need the following bound on Δ_j^+.

Lemma 2.2. *For every priority j, $\Delta_j^+ < f(j+1)/f(j)$.*

Proof. Let s be the first stage at which some root $v \in R_j$ has outdegree Δ_j^+ in A_j. Then there exists another root $w \in R_j$ such that $v \to w$ enters D_j at stage s. Let x be the new (nonroot) that is presented at stage s. Then the priority of x must be j and the root of x must be v. Thus $d(v,x) \leq R(j) + l$. Also there exists another nonroot y with priority j such that $x \sim y$ and w is the root of y.

We have chosen the functions R and r to satisfy the following recursive inequalities.

(a) $r(j) + r(j+1) + 3R(j) + 4 \leq R(j+1)$ and

(b) $r(j) + 2R(j) + 3 \leq r(j+1)$.

Claim 1. If $v \to u$ in A_j at stage s, then the priority of u at stage s is j.

Proof. Since u is a vertex in A_j at stage s, the priority of u is at least j at stage s. At stage s

$$d(u,x) \leq d(u,v) + d(v,x) \leq 2R(j) + 3 + R(j) + 1 \leq 3R(j) + 4 \leq R(j+1).$$

Thus, if the priority of u at stage s were larger than j, then x would be in $N(S(u))$ and the priority of x would have to be at least that of u, a contradiction. ∎

Claim 2. Suppose that at stage s, $v \to u$ in A_j and r is a root with priority $i > j$. Then $W(r) \cap W(u) = \emptyset$

Proof. Suppose that $z \in W(r) \cap W(u)$ at stage s. Then

$$d(r,x) \leq d(r,z) + d(z,u) + d(u,v) + d(v,x) \leq r(i) + r(j) + 2R(j) + 3 + R(j) + 1$$
$$\leq r(i) + r(j) + 3R(j) + 4 \leq R(i).$$

Then $x \in N(S(u))$ and thus the priority of x is at least i, a contradiction. ∎

Let $W = \cup\{W(u) : v \to u \text{ in } A_j \text{ at stage } s\}$. By Claim 1 and the fact that roots with the same priority have disjoint witness sets, $|W| \geq \Delta_j f(j)$. By Claim 2, W does not intersect the witness set of any root with priority greater than j. Suppose $z \in W$ and z has root u, with $v \to u$ in A_j at stage s. Then

$$d(z,v) \leq d(z,u) + d(u,v) \leq r(j) + 2R(j) + 3 \leq r(j+1).$$

Thus if $|W|$ were at least $f(j+1)$, W would witnesses that the priority of v at stage s is at least $j+1$. We conclude that $|W|<f(j+1)$. Thus $\Delta_j^+ < f(j+1)/f(j)$. ∎

Using Lemma 2.2 we obtain the following upper bound on the number of global colors used:

$$G(j) \le f(j+1)^2/f(j)^2 \le n^{8/\log\log n}. \tag{5}$$

Combining (1), (2), (4) and (5) we get

$$a(k,n) \le \frac{1}{4} n^{1+10k-9+8)/\log\log n} = \frac{1}{4} n^{10k/\log\log n}.$$ ∎

We can now easily obtain our main result from Theorem 2.1.

Proof of Theorem 0.1. Fix k and let $n_1=1$. Let $f(k,n)=\frac{1}{4}n^{10k/\log\log n}$. For any positive integer $i>1$, define n_i recursively by $n_i=\min\{n: 2f(k,n_{i-1}) \le f(k,n)\}$. Then $f(k,n_i-1)<2f(k,n_{i-1}) \le f(k,n_i)$. Let $\mathbf{A}_k$ be the following on-line algorithm. For an on-line graph $G^<$ with $V(G)=\{x_1<\ldots<x_n\}$, $\mathbf{A}_k$ assigns the vertex x_j the color (i,c), where $n_{i-1}<j<n_i$ and c is the color that $\mathbf{A}_{k,n_i}$ assigns x_j when considered as a vertex of the on-line induced subgraph $G^<[\{x_{n_{i-1}}+1,\ldots,x_{n_i}\}]$. Let $b(k,n)$ be the number of colors that $\mathbf{A}_k$ uses to color an on-line k-colorable perfect graph on n vertices.

Claim. For all i, $b(k,n_i)<2f(k,n_i)$.

Proof. We argue by induction on i. The base step $i=1$, follows from the observation that, since $n_1=1$, $b(k,n_1)=1=f(k,n_1)$. For the induction step $i>1$, note that

$$b(k,n_i) < b(k,n_{i-1}) + f(k,n_i) < 2f(k,n_{i-1}) + f(k,n_i) \le 2f(k,n_i).$$

∎

Suppose that $n_i<n<n_{i+1}$. Using the claim, we have

$$b(k,n) \le 2f(k,n_i) + f(k,n_{i+1}-1) \le 4f(k,n_i) \le 4f(k,n).$$

Thus $b(k,n) \le n^{10k/\log\log n}$. ∎

3. The lower bound

In this section we prove Theorem 0.2. Vishwanathan proved that for every randomized on-line algorithm $\mathbf{A}$, there exists an on-line k-colorable graph $G^<$ such that the expected value of $\chi_{\mathbf{A}}(G^<)$ satisfies

$$E[\chi_{\mathbf{A}}(G^<)] \le \frac{1}{k-1}\left(\frac{-1+\lg n}{2k+1}\right)^{k-1}.$$

We shall show that Vishwanathan's construction actually yields perfect graphs. Taking into account the conflicting interests of completeness and brevity, we shall only consider the case of deterministic on-line algorithms. This allows us to present our modification in a simpler setting and to obtain a slightly stronger bound. The interested reader can then apply the same technique to the randomized version.

We shall need the following additional notation. Let $G^{<}$ and $H^{\ll}$ be on-line graphs.

Then $G+H$ denotes the graph obtained by taking the disjoint union of G and H. $G^{<}+H^{\ll}$ denotes the on-line graph obtained from $G+H$ by first presenting every vertex of G in the order $<$ and then presenting every vertex of H in the order $\ll$. If E is a set of edges on the vertices of G, then $G^{<}+E$ is the on-line graph obtained by adding the edges of E to G and presenting the vertices of G in the order $<$. Consider an on-line algorithm $\mathbf{A}$, an on-line graph $G^{<}$, and a subset X of $V(G)$. Then there exists an on-line algorithm $\mathbf{A}(G^{<},X)$ such that for any on-line graph $H^{\ll}$, the color that $\mathbf{A}(G^{<},X)$ assigns any vertex v of $H^{\ll}$ while coloring $H^{\ll}$ is the same as the color that $\mathbf{A}$ assigns v while coloring $G^{<}+H^{\ll}+\{xy : x \in X \quad \text{and} \quad y \in V(H)\}$. The algorithm $A(G^{<},X)$ simply colors $H^{\ll}$ by applying $\mathbf{A}$ to the on-line graph $G^{<}+H^{\ll}+\{xy : x \in X \quad \text{and} y \in V(H)\}$.

Proof of Theorem 0.2. Let $g(k,n)=\left(\frac{\log_3 n}{2k}\right)^{k-1}$. Suppose that s and k are positive integers and $n=s^k$. We shall prove that for every on-line algorithm $\mathbf{A}$, there exists an on-line k-colorable perfect graph $G^{<}$ on n vertices such that $\chi_{\mathbf{A}}(G^{<})\geq g(k,n)$. The theorem then follows from a short calculation and the observation that any positive integer n satisfies $3^S \leq n < 3^{S+1}$, for some positive integer s.

Following Vishwanathan [11] we shall argue by a double induction with the primary induction on k and the secondary induction on s. The key idea of Vishwanathan was to maintain the stronger induction hypothesis that there exists an on-line k-colorable graph $G^{<}$ such that $\mathbf{A}$ uses many colors on one of the k color classes of some (off-line) k-coloring of G. To insure that G is perfect, we must strengthen this hypothesis as in (3) below. We say that a pair $(G^{<},I)$ consisting of an on-line graph $G^{<}$ and an independent subset I of G is a (k,n)-*witness* for an on-line algorithm $\mathbf{A}$ if

(1) $G^{<}$ has at most n vertices;

(2) $\omega(G)\leq k$;

(3) for all induced subgraphs $H=G[W]$ of G, H can be $\omega(H)$-colored so that $I\cap W$ is contained in a color class; and

(4) $\mathbf{A}$ uses at least $g(k,n)$ colors on the vertices of I.

Notice that (3) insures that G is perfect and thus (2) insures that G is k-colorable. Of course (4) insures that $\mathbf{A}$ uses at least $g(k,n)$ colors on $G^{<}$. Thus it suffices to prove that for every on-line algorithm $\mathbf{A}$ there exists a (k,n)-witness for $\mathbf{A}$. The primary base step $k=1$ is trivial since $g(1,n)=1$, so consider the primary inductive step $k>1$. Also the secondary base step $s=0$ is trivial since $g(k,1)\leq l$.

For the secondary induction step $t=s+1$, i.e, $m=3n$, we must show that for every on-line algorithm **A**, there exists a $(k,3n)$-witness for **A**. We first check that g satisfies the following recursive inequality:

(5) $g(k,3n)\le g(k,n)+\frac{1}{2}g(k-1,n)$, which is equivalent to
$(2k)^{k-1}(g(k,3n)-g(k,n))\le(2k)^{k-1}\frac{1}{2}g(k-1,n)$.
Writing $x=log_3 n$, and using the fact that $k-1\le x$, we have

$$\begin{aligned}(2k)^{k-1}(g(k,3n)-g(k,n)) &= (x+1)^{k-1}-x^{k-1}\\ &= \sum_{i=1}^{k-1}\binom{k-1}{i}x^{k-1-i}\\ &\le \sum_{i=1}^{k-1}\frac{(k-1)^i}{i!}x^{k-1-i}\\ &\le (k-1)x^{k-2}(e-1)\end{aligned}$$

On the other hand,

$$(2k)^{k-1}\frac{1}{2}g(k-1,n)=k\left(\frac{k}{k-1}\right)^{k-2}x^{k-2}.$$

Now we are done since $e-1\le\left(\frac{k}{k-1}\right)^{k-1}$, for all $k>1$.

Fix an on-line algorithm **A**. We must construct a $(k,3n)$-witness for **A**. Using the primary and secondary induction hypotheses, we can find pairs $(G_i^<,I_i)$, $i\in\{1,2,3\}$ such that

(6) $(G_1^<,I_1)$ is a (k,n)-witness for $\mathbf{A}_1=\mathbf{A}$;

(7) $(G_2^<,I_2)$ is a (k,n)-witness for $\mathbf{A}_2=\mathbf{A}(G_1^<,\emptyset)$; and

(8) $(G_3^<,I_3)$ is a $(k-1,n)$-witness for $\mathbf{A}_3=\mathbf{A}(G_1^<+G_2^<,I_1)$.

Let $G^<=G_1^<+G_2^<+G_3^<+\{xy:x\in I_1 \text{ and } y\in V(G_3)\}$. Note that the color that **A** assigns a vertex $v\in V(G_i)$ while coloring $G^<$ is the same as the color $\mathbf{A}_i$ assigns v while coloring $G_i^<$. We shall show using (5) that either $(G^<,I_1\cup I_2)$ or $(G^<,I_3\cup I_2)$ is a $(k,3n)$-witness for **A**. Clearly (1) and (2) hold in either case. Next we check that (3) holds for $I_1\cup I_2$ and $I_3\cup I_2$. Assume that $I_1\cap W\neq\emptyset$, since otherwise this is trivial. Let $H=G[W]$ be an induced subgraph of G. For $i\in\{1,2,3\}$, let $H_i=G[W\cap V(G_i)]$. By (6)–(8) there exists an $\omega(H_i)$-coloring f_i of H_i such that $W\cap I_i$ is contained in a color class, for all $i\in\{1,2,3\}$. We may assume that range$(f_1)=[\omega(H_1)]$, range$(f_2)\subset[\omega(H)]$, and range$(f_3)=\{2,\dots,\omega(H_3)+1\}$. We may also assume that f_1 colors I_1 with the color 1 and f_3 colors I_3 with the color 2. Since $I_1\cap W\neq\emptyset$, $\omega(H_3)+1\le\omega(H)$, and thus $f=f_1\cup f_2\cup f_3$ is an $\omega(H)$-coloring of H. If we want $I_1\cup I_2$ contained in a color class of f, we choose f_2 so that f_2 uses color 1 on I_2. If we want $I_3 \cup I_2$ contained in a color class, we choose f_2 so that f_2 uses the color 2 on I_2.

Finally we check (4) holds for either $I = I_1 \cup I_2$ or $I = I_3 \cup I_2$. For $i \in \{1,2,3\}$, let C_i be the set of colors that **A** uses on I_i when coloring $G^<$. Since $(G_1^<, I_1)$ and $(G_2^<, I_2)$ are (k,n)-witnesses and $(G_3^<, I_3)$ is a $(k-1,n)$-witness, $|C_1|, |C_2| \geq g(k,n)$ and $|C_3| \geq g(k-1,n)$ Notice that $C_1 \cap C_3 = \emptyset$, since every vertex in G_3 is adjacent to every vertex in I_1. If $|C_2 \cap C_3| \geq \frac{1}{2} g(k-1,n)$, then $|C_1 \cup C_2| \geq g(k,n) + \frac{1}{2} g(k-1,n) \geq g(k,3n)$ by (5); thus (4) holds for $I = I_1 \cup I_2$. Otherwise $|C_2 \cap C_3| \leq \frac{1}{2} g(k-1,n)$, and so $|C_2 \cup C_3| \geq g(k,n) + \frac{1}{2} g(k-1,n) \geq g(k,3n)$. Thus (4) holds for $I = I_2 \cup I_3$. This completes the proof. ∎

4. Graphs without induced odd cycles

András Gyárfás pointed out that the crux of our arguments does not depend on the graph being perfect, but only on it having no induced odd cycle on more than three vertices. This observation leads to the following Theorem.

Theorem 4.1. *There exists an on-line algorithm* **A** *that will color any on-line k-colorable graph $G^<$ on n vertices with clique size ω that does not induce an odd cycle on more than three vertices with at most $n^{\omega\left(\frac{9}{\log\log n} + \frac{\log k}{\log^{1/2} n}\right)}$ colors.*

Sketch of Proof. It is easy to check that the proofs of both Lemma 1.1 and Corollary 1.2 apply to graphs that do not induce an odd cycle on more than three vertices. Again $\mathbf{A}(\omega, k, n)$ is constructed by recursion on ω. The only change in the number of colors used occurs in selecting the sphere colorings. These colorings must now use k colors instead of ω colors. This accounts for the $\log k / \log^{1/2} n$ tein the exponent, which was previously estimated in (3) by $1/\log\log n$.

Acknowledgments. We are grateful to an anonymous referee whose suggestions greatly improved the presentation of this paper. We also thank András Gyárfás for pointing out Theorem 4.1.

References

[1] D. Bean: Effective coloration, *J. Symbolic Logic* **41** (1976), 469–480.

[2] A. Gyárfás, and J. Lehel: On-line and first-fit coloring of graphs, *J. of Graph Theory* **12** (1988), 217–227.

[3] A. Gyárfás, and J. Lehel: First-Fit and on-line chromatic number of families of graphs, *Ars Combinatorica* **29C** (1990), 168–176.

[4] S. IRANI: Coloring inductive graphs on-line, *Proceedings of the 31st Annual Symposium on the Foundations of Computer Science,* (1990), 470–479.

[5] H. A. KIERSTEAD: The linearity of First-Fit for coloring interval graphs, *SIAM J. on Discrete Math.,* **1** (1988), 526–530.

[6] 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 J. on Discrete Mathematics,* **8** (1995), 485–498.

[7] H. A. KIERSTEAD, S. G. PENRICE, and W.T. TROTTER: On-line graph coloring and recursive graph theory, *SIAM J. on Discrete Math.* **7** (1994), 72–89.

[8] H. A. KIERSTEAD, and W. T. TROTTER: An extremal problem in recursive combinatorics, *Congressus Numerantium,* **33** (1981), 143–153.

[9] L. LOVÁSZ, M. SAKS, and W. T. TROTTER: An online graph coloring algorithm with sublinear performance ratio, *Discrete Math.,* (1989), 319–325.

[10] M. SZEGEDY: Private communication.

[11] S. VISHWANATHAN: Randomized online graph coloring, *J. Algorithms,* **13** (1992), 657–669.

H. A. Kierstead

Department of Mathematics
Arizona State University,
Tempe AZ 85287, U.S.A.
kierstead@math.la.asu.edu

K. Kolossa

Department of Mathematics
Arizona State University,
Tempe AZ 85287, U.S.A.
askxk@asuvm.inve.asu.edu