Available online at www.sciencedirect.com
ScienceDirect
Theoretical Computer Science 363 (2006) 18–27
Theoretical Computer Science
www.elsevier.com/locate/tcs

An improved algorithm for online coloring of intervals with bandwidth[☆]

Yossi Azar[a,1], Amos Fiat[a,1], Meital Levy[a], N.S. Narayanaswamy[b,*]

[a] School of Computer Science, Tel-Aviv University, Israel
[b] Department of Computer Science and Engineering, Indian Institute of Technology Madras, Chennai 600 036, India

Abstract

We present an improved online algorithm for coloring interval graphs with bandwidth. This problem has recently been studied by Adamy and Erlebach and a 195-competitive online strategy has been presented. We improve this by presenting a 10-competitive strategy. To achieve this result, we use variants of an optimal online coloring algorithm due to Kierstead and Trotter.

Keywords: Online algorithms; Graph coloring; Interval graphs

1. Introduction

Interval graphs are of interest as they are used to model the structure of many problems in a variety of fields. One classical well-studied problem is the issue of vertex coloring of interval graphs. Features of containment associated with intervals are well exploited to design efficient algorithms and fast implementations to optimally color interval graphs (see [6]). Interval graphs are used to model many resource allocation problems (see book [6]). Each interval corresponds to a request for the usage of a resource exclusively for a certain period of time. In this paper we consider the issue of online coloring a set of intervals based on some relaxed properties (two adjacent vertices may get the same color as long as an additional condition is not violated).

Online algorithms are motivated by environments where the inputs occur in a sequence and need to be serviced without any knowledge of the future inputs. Online interval coloring algorithms are one example of many such algorithms that are of much interest (see book [3]). The problem of coloring interval graphs with bandwidths (CIB) is a generalization of the interval graph coloring problem (CI). In the generalization, each interval has a bandwidth in $(0, 1]$. These bandwidths are referred to as bandwidths. A valid coloring is one that satisfies the condition that, for every r on the

☆ A preliminary version of the results in this paper were presented at COCOON 2004 [11], and at OLA 2004 [2].
* Corresponding author.
E-mail addresses: azar@post.tau.ac.il (Y. Azar), fiat@tau.ac.il (A. Fiat), levymeit@post.tau.ac.il (M. Levy), swamy@shiva.iitm.ernet.in (N.S. Narayanaswamy).
[1] Research supported in part by the German–Israeli Foundation.

0304-3975/$ - see front matter
doi:10.1016/j.tcs.2006.06.014

real line, for every color c, the set of intervals colored c, containing r have bandwidth that sum up to at most 1. Clearly, when each interval has bandwidth 1, we get the interval graph coloring problem. We refer to a coloring satisfying the above condition as a *bandwidth satisfying coloring*. A coloring that assigns different colors to the end points of each edge is simply referred to as a *valid coloring*.

Online coloring of intervals with bandwidth has been of recent interest in [1] and, as been remarked there, is a simultaneous generalization of two other problems:

(1) Online Bin Packing, the study of which dates back to work of Johnson in the early 1970s [7]. If all intervals have the same left and right endpoints would correspond to Bin Packing where each color represents a bin.
(2) Online Coloring of Interval graphs (CI), introduced by Kierstead and Trotter [10].

Note: Online coloring of intervals and online coloring of interval graphs are one and the same. An interval corresponds to a vertex and there is an edge between two vertices if and only if the corresponding intervals intersect. The chromatic number of the interval graph is the same as the maximum clique size which is attained at a point where the largest number of intervals intersect.

Motivation: CIB is motivated by many applications. One example is that of a network with line topology that consists of links, where each link has channels with constant capacity (all the channels have the same capacity c). The channels can be either an all-optical WDM (wavelength-division multiplexing) network or an optical network supporting SDM (space-division multiplexing). The connection requests between two points, say from a to b, request bandwidth, and the total requests assigned to a channel must not exceed the capacity of the channel on any of the links that connect a and b.

Previous work: Online coloring algorithms for interval graphs have been well studied. The best known algorithm is in [8,10] which uses at most $3\omega - 2$ colors where ω denotes the maximum clique size in the interval graph. It is also shown to be an optimal online algorithm in [10]. Another approach to solving online coloring is to use the First fit: allot the smallest valid color to each vertex. Much research has been done analyzing the simple First Fit algorithm for the CI problem. An upper bound of 40 on the competitive ratio was given first by Kierstead, and later improved to 25.72 by Kierstead and Qin [9]. Lower bounds for First Fit was also studied and Chrobak and Slusarek [4] obtain a constant lower bound between 4.4 and 4.5. Coloring intervals with bandwidth (CIB) was introduced by Adamy and Erlebach [1]. Their work uses a combination of the optimal online algorithm and the First Fit approach. This algorithm was shown to achieve a constant competitive ratio of 195 [1]. Subsequently, an improved analysis, by Pemmaraju and Raman [12], shows that the competitive ratio of the Adamy–Erlebach algorithm is 35. As argued in [1] First Fit for CIB can perform arbitrarily bad. They also show that First Fit gets a competitive ratio of 192 if all intervals have bandwidth $\leqslant 1/2$.

1.1. Our results

Our results for the CIB follow by a careful adaptation of an online algorithm due to Kierstead and Trotter [10] for coloring interval graphs.

Upper and lower bounds for CIB: We give an online algorithm for the CIB problem with a competitive ratio of 10. The algorithm is strongly influenced by the 3-competitive algorithm for CI by Kierstead and Trotter [10]. We design a new online algorithm to color intervals with bandwidth. Our approach is similar to that in [1] where the requests are classified into two classes based on the bandwidth. In [1] one subproblem is colored using the first fit, and the other using the optimal online coloring algorithm. Each subproblem uses a distinct set of colors.

In our approach, we classify the input into three classes, again based on bandwidth. We then appropriately apply variants of the optimal online algorithm to each class, operating on a different set of colors for each class. The online algorithm is modified to suit the needs of each class, and this modification is what leads to an improved performance. We present online coloring algorithms for intervals with bandwidth in different ranges. We give a $\frac{2m}{m-2}$ competitive ratio for coloring intervals with bandwidth in the range of $\left[0, \frac{1}{m}\right]$, m integer $\geqslant 3$. We show an online coloring algorithm for intervals with bandwidth in the range of $\left[\frac{1}{m}, \frac{1}{2}\right]$ with a competitive ratio of $(m-1)$. The third class is formed by those intervals with bandwidths in the range $\left(\frac{1}{2}, 1\right]$. On this class we use the optimal algorithm for online coloring intervals graphs which guarantees a competitive ratio of 3.

Lower bounds: We give a lower bound of $\frac{4}{3}$ on the competitive ratio of any online algorithm for intervals with bandwidth in $[a, b]$ for any $0 \leqslant a, b \leqslant 1$. This means that even if all intervals have very small bandwidth the lower bound is strictly greater than 1.

Outline of paper: In Section 2 we present the preliminaries. In Section 3 the 10 competitive algorithm is presented. In Section 4 we present the lower bound for small bandwidth intervals. Finally, we present some conclusions and open problems in Section 5.

2. Preliminaries

For a set of intervals in the real line $P = \{I_i : 1 \leqslant i \leqslant n\}$, we can associate a graph denoted by $G(P)$. In this graph, there are as many vertices as the number of intervals in P. Each interval is associated with a unique vertex and vice versa. Two vertices are adjacent if there is a non-empty intersection between the corresponding intervals. Throughout the paper, we use the words interval and vertex interchangeably for ease of presentation. In particular, when we deal with properties of the underlying interval graph we think of the intervals as vertices.

For an undirected graph G, $\omega(G)$ denotes the size of the maximum cardinality clique in G. $\Delta(G)$ denotes the $\max\{\text{degree of } u : u \in V(G)\}$. For a weighted graph G, which is a graph where each vertex has a weight, $\omega^*(G)$ denotes the largest weighted clique in G. The size of weighted clique is the sum of the weights of the vertices in the clique. Let P be a collection of intervals. We now define three notions of *density* with respect to the collection P. For a positive integer r, the *density* of r is defined to be $D(r) = |\{I \in P : r \in I\}|$. The density of an interval I is defined as $D(I) = \min\{D(r) : r \in I\}$. The density of P, $D(P)$, is defined as $\max\{D(I) : I \in P\}$. We present the following crucial lemma from [8] for the sake of clarity.

Lemma 2.1. *Let P be a collection of intervals and $G = G(P)$. If $D(P) = 1$ then $\omega(G) \leqslant 2$ and $\Delta(G) \leqslant 2$.*

Proof. Since $D(P) = 1$, every interval I has a point r such that the point is present exclusively in I. I.e., $D(I) = 1$ for every $I \in P$. Therefore, if $\omega(G) \geqslant 3$ then there would exist 3 intervals I_1, I_2, I_3 corresponding to vertices in G which form a triangle (a 3-clique). In any three such intervals, one of them, say I_1, is contained in the union of the other two. Each point of I_1 is contained in either I_2 or I_3 (some point are contained in both I_2 and I_3, but we do not need this fact here). It follows that $D(I_1) \geqslant 2$. This contradicts the hypothesis that $D(P) = 1$. Therefore, $\omega(G) \leqslant 2$. Similarly, a vertex of degree at least 3 implies that there is an interval which is contained in the union of at most two other intervals. This would violate the hypothesis that $D(P) = 1$. Therefore, $\Delta(G) \leqslant 2$. □

2.1. Online coloring interval graphs

It is well known that an interval graph can be colored optimally with as many colors as the size of its largest clique. The algorithm first orders the vertices according to the leftmost point of the intervals associated with them. In other words, the first vertex in the order is one whose leftmost point is the smallest. Then the vertices are considered according to the constructed order and greedily colored: every interval is assigned to the smallest valid color. The coloring problem for interval graphs becomes more challenging when we consider the problem of designing online algorithms. Here, along with the input interval graph, an order $\sigma = v_1, \ldots, v_n$ is presented. The algorithm must color the vertices according to σ and use as few colors as possible. Below we present the online coloring algorithm due to Kierstead and Trotter for interval graphs which uses at most $3\omega - 2$ colors. The algorithm partitions the vertices into sets, and every set is colored with 3 colors, using a different set of colors for each set. The algorithm can also be visualized as running in two phases: in the first phase an arrangement of the intervals satisfying certain properties is computed. In the second, the arrangement is 3-colored to obtain a coloring of the interval graph.

Formally, let v_i be the current vertex to be colored. Below, we identify a position, $p(v_i)$, for v_i depending on $p(v_1), \ldots, p(v_{i-1})$. To decide the value of $p(v_i)$ we consider the following graph which is defined for each integer $k \geqslant 0$: $G_k(v_i)$ is the induced subgraph of G on the vertex set $\{v_j \in V(G) : j < i, p(v_j) \leqslant k, \{v_i, v_j\} \in E(G)\}$.

Theorem 2.2 (*Kierstead and Trotter [10]*). *Algorithm* 1 *uses* $3w - 2$ *colors where* w *is the max cardinality of the clique.*

The proof of Kierstead and Trotter is based on the following properties:
Properties of $p(v)$:

Algorithm 1. Online coloring interval graphs.

A_{KT}(Interval v_i)

$p(v_i) =$ the smallest non-negative integer r such that $\omega(G_r(v_i)) \leqslant r$.
Let $SameP(v_i) = \{v | p(v) = p(v_i)\}$
$color(v_i) = f \times p(v_i)$ such that no element of $SameP(v_i)$ has been assigned color $f \times p(v_i)$ where $f \in \{1, 2, 3\}$.

1. For each v, $p(v) \leqslant \omega - 1$.
2. For a number j, consider the collection P of intervals corresponding to the vertices of the induced graph on $\{v | p(v) = j\}$. This collection has density equal to 1. From Lemma 2.1 it follows that the maximum vertex degree in this graph is at most 2.
3. $p(v_i)$ depends only on the vertices which were considered prior to v_i and $color(v_i)$ depends on the color of at most two of its neighbors *on* the line $p(v_i)$. Further, vertices with $p(v) = 0$, all get the same color, as they form an independent set. Consequently, it follows that the two phases can be performed online and at most $3\omega - 2$ colors are used.

3. Upper bounds

Recall that the bandwidth of each interval is a number in (0, 1]. The goal is to use the minimum number of colors to color the vertices of an interval graph such that, for each color c, the weight of the maximum weight clique is bounded by 1 in the graph induced by the vertices assigned c. The algorithm should be an online strategy, in the sense that at each time step (decided by the algorithm), an input pair consisting of an interval and its bandwidth requirement is presented to the algorithm. This request should be serviced by assigning a color before the next request can be presented.

3.1. Upper bound for the case of small bandwidths

Algorithm 2. A_{Small}.

A_{Small}(Integer m $\geqslant 3$, Interval I)

ω^* = Weight of a maximum clique on all intervals processed thus far, including I.
$j = \left\lceil \omega^* \cdot \frac{2m}{m-2} \right\rceil$;
pass I to A_j;

Algorithm 3. A_j.

A_j
Init /*First call*/
$B_j = B_{j-1} \cup C_{j-1}$;
$C_j = \emptyset$;
For a new interval I,

if $\omega^*(B_j \cup \{I\}) \leqslant (j-1)\left(\frac{1}{2} - \frac{1}{m}\right)$ **then**
  $B_j = B_j \cup I$;
  pass I to A_{j-1};
else
  $C_j = C_j \cup I$;
  Color I with the color j;
end if

In this section we consider the case when the bandwidths are in the range $\left[0, \frac{1}{m}\right]$, $m \geqslant 3$. To handle this case we present the algorithm, A_{Small}, that will be shown in Theorem 3.1 to be 2 competitive asymptotically.

Algorithm A_j. Formally, for each $j \geqslant 0$, A_j is associated with two sets of intervals, B_j and C_j. When A_j is first used, the sets B_j and C_j are defined, recursively, as follows:

- $B_j = B_{j-1} \cup C_{j-1}$;
- $C_j = \emptyset$.

Subsequently, A_j maintains an online partition of the intervals into B_j and C_j. When a new interval I is presented, A_j puts I into one of two sets: A_j puts I into the set B_j, if $\omega^*(B_j \cup \{I\}) \leqslant (j-1)\left(\frac{1}{2} - \frac{1}{m}\right)$, otherwise, A_j puts I into C_j. If I is put into the set B_j, it is colored by A_{j-1} . Otherwise, it is colored j in the set C_j. Note that for every j, C_j contains the set of all intervals that were colored by A_j with j and B_j contains the set of all intervals that were colored by a color smaller than j.

The main property of the sub algorithm A_j is that it uses one color, namely j. Therefore, algorithms $A_1, A_2, \ldots, A_j$ use a total of j colors. The value j, at any point in time, depends on the maximum total bandwidth of intersecting intervals presented to the algorithm up to that time instant. Recall that the sum of bandwidths of intersecting intervals is the weight of a clique in the corresponding weighted interval graph. The algorithm A_j is used as long as the maximum total bandwidth among a set of intervals *maintained* by A_j, denoted by k, is below $j\left(\frac{1}{2} - \frac{1}{m}\right)$. Since we deal with an online process, k increases over time. When k increases to more than $j\left(\frac{1}{2} - \frac{1}{m}\right)$, then A_{Small} can no longer use A_j to color the newly presented interval, so it uses the next algorithm A_{j+1}.

Theorem 3.1. *A_{Small} is $\frac{2m}{m-2}$ competitive.*

Analysis: Clearly, the above scheme is an online strategy, as it processes a request fully before considering the next. Our interest is in the number of colors that are used by the strategy, and the correctness of the result. The number of colors used is at most $\frac{2m}{m-2}$ times the optimum. The reason is that any bandwidth satisfying assignment uses at least $\lceil \omega^* \rceil$ colors, and our algorithm above uses $j = \left\lceil \omega^* \cdot \frac{2m}{m-2} \right\rceil$ colors.

Correctness: We are left to prove that every set C_j, can be colored by one color only. We next show that for each j the weight of any clique in C_j is at most 1.

Let t be the time interval I had been given a color. Denote by $B_{(j,t)}$ and $C_{(j,t)}$ be the sets B_j and C_j at time t, respectively.

A *critical point* p in interval $I \in C_{(j,t)}$ colored at time t, is a point with a total bandwidth of more than $(j-1)\left(\frac{1}{2} - \frac{1}{m}\right)$ in $B_{(j,t)} \cup C_{(j,t)}$. For every colored interval there is at least one critical point.

Lemma 3.2. *For every $I \in C_{(j,t)}$ and every critical point $p \in I$, where I is colored at time t, the total bandwidth at p of intervals in $C_{(j,t)}$ does not exceed $\frac{1}{2}$ for every j and t.*

Proof. Proof by contradiction; assume that there is a critical point, $p \in I \in C_{(j,t)}$, where the weighted clique of intervals in $C_{(j,t)}$ is more than $\frac{1}{2}$. Since p is a critical point, by definition $\omega^*(B_{(j,t)} \cup I) > (j-1)\left(\frac{1}{2} - \frac{1}{m}\right)$ at point p. Since $b(I) \leqslant \frac{1}{m}$, the weighted clique of $B_{(j,t)}$ at p is greater than $(j-1)\left(\frac{1}{2} - \frac{1}{m}\right) - \frac{1}{m}$. Since $\omega^*(C_{(j,t)}) > \frac{1}{2}$ in p we get that $\omega^*(B_{(j,t)} \cup C_{(j,t)}) > (j-1)\left(\frac{1}{2} - \frac{1}{m}\right) - \frac{1}{m} + \frac{1}{2} = j\left(\frac{1}{2} - \frac{1}{m}\right)$. But this contradicts the property that A_j only deals with intervals with a total bandwidth that does not exceed $j\left(\frac{1}{2} - \frac{1}{m}\right)$. Therefore, our assumption is false. Hence the lemma holds. $\square$

Lemma 3.3. *For every j and t, $\omega^*(C_{(j,t)}) \leqslant 1$.*

Proof. Proof by contradiction, assume that there is a weighted clique of more than 1 in $C_{(j,t)}$ at point p_j. But this means that either the first critical point to the left of p_j, or the first critical point to the right of p_j, has a total bandwidth of more than $\frac{1}{2}$. By Lemma 3.2, this is not possible. $\square$

Taking t to be the time the last interval is colored, it follows that the algorithm produces a bandwidth satisfying coloring.

This concludes the proof of Theorem 3.1.

Remark 1. A_{Small} can be modified reducing slightly the number of colors used to $\left\lceil \omega^* \cdot \frac{2m}{m-2} \right\rceil - \left(\left\lfloor \frac{2m}{m-2} \right\rfloor - 1 \right)$. We let A_j use the first color only for integer values of $j \leqslant \left(\frac{2m}{m-2} - 1 \right)$. This can be done since $\left(\frac{2m}{m-2} - 1 \right) \left(\frac{1}{2} - \frac{1}{m} \right) \leqslant 1$. This means that the weight of no clique in $B_j \cup C_j$ for $j \leqslant \left(\frac{2m}{m-2} - 1 \right)$ exceeds 1.

Remark 2. Note that we have proved that algorithms in a larger family (which include A_{Small}) are $\frac{2m}{m-2}$ competitive. The family contains all algorithms such that, given an interval I find some j satisfying $\omega^*(B_{j+1} \cup \{I\}) \leqslant j \left(\frac{1}{2} - \frac{1}{m} \right)$ and $\omega^*(B_j \cup \{I\}) > (j-1) \left(\frac{1}{2} - \frac{1}{m} \right)$ and colors I with j.

3.2. Upper bound for the case of middle bandwidth

Let $\sigma = v_1, \ldots, v_n$ be the ordering of the vertices of G. Here, we consider the case where for every interval v_i, $\frac{1}{m} < b(v_i) \leqslant \frac{1}{2}$. The coloring strategy described below uses as many colors as the size of the largest unweighed clique, ω formed by intervals whose bandwidth is in $\left(\frac{1}{m}, \frac{1}{2} \right]$. The color of v_i, denoted by $color(v_i)$, is decided based on $color(v_1), \ldots, color(v_{i-1})$. To identify the color of v_i, we use the following graph which is defined for each integer $k \geqslant 0$: $G_k(v_i)$ is the induced subgraph of G on the vertex set $\{v_j \in V(G) : j < i, color(v_j) \leqslant k, \{v_i, v_j\} \in E(G)\}$.

Algorithm 4. Online coloring interval graphs with bandwidths in $\left(\frac{1}{m}, \frac{1}{2} \right]$.

A_{middle}(Integer m, Interval v_i)
1: $color(v_i)$ = the smallest non-negative integer r such that $\omega(G_r(v_i)) \leqslant r$.

In other words, the above algorithm ignores the bandwidth requirements, i.e., treats all these bandwidths as 1, and applies only the first two steps of Algorithm 1. Exactly ω colors are used to color the vertices based on this arrangement. As the bandwidths are at most $\frac{1}{2}$, we have obtained a bandwidth satisfying coloring. Recall that ω denotes the size of the largest clique in the graph induced by the intervals in question.

Lemma 3.4. *The above strategy outputs a bandwidth satisfying coloring using at most ω colors.*

Proof. As argued in Section 2.1, for each $y \in \{0, 1, \ldots, \omega - 1\}$, the interval graph formed by intervals in the set $\{v | color(v) = y\}$ does not have a clique with more than 2 vertices. Further, for each v, $color(v) \leqslant \omega - 1$. Clearly, at most ω colors are used, and the coloring is bandwidth satisfying as each bandwidth requirement is at most $\frac{1}{2}$. Hence the lemma is proved. □

To compare with the optimum bandwidth satisfying assignment, we give a lower bound on the number of colors in an optimum bandwidth satisfying coloring in terms of the number of colors in an optimum valid coloring.

Lemma 3.5. *The number of colors used by an optimum bandwidth satisfying coloring is at least $\frac{1}{m-1}$ times the number of colors used in an optimum valid coloring which is at least ω.*

Proof. In any optimum bandwidth satisfying solution, the largest clique size among monochromatic vertices is $m-1$. If it exceeds m, then such a solution would not be feasible, as the clique weight would be more than 1 due to the bandwidths being more than $\frac{1}{m}$. To obtain a valid coloring, we optimally color each color class using at most $m-1$ colors. The number of colors used is at most $m-1$ times the optimum bandwidth satisfying coloring. Therefore, the optimum valid coloring is at most $(m-1)$ times the optimum bandwidth satisfying coloring. In other words, an optimum bandwidth satisfying coloring uses at least $\frac{1}{m-1}$ times the number of colors used by an optimum valid coloring. □

Number of colors used: From the previous two lemmas it follows that the online algorithm uses at most $m-1$ times the number of colors used by an optimum bandwidth satisfying coloring. The above result is a specific case of a more general algorithm called STEADY in [5]. Consequently, the algorithm is $(m-1)$ competitive.

3.3. Upper bound for the case of high bandwidth

Here, we consider the case where every interval has bandwidth of strictly higher than $\frac{1}{2}$.

Algorithm A_{KT}. The algorithm is to simply color the intervals ignoring the bandwidth requirements. As in [1] we apply Kierstead and Trotter's online algorithm to color each interval. From the analysis of [10] it follows that the number of colors used is at most 3 times the optimum number of colors.

Theorem 3.6. *A bandwidth satisfying coloring output by A_{KT} uses at most three times the optimum number of colors.*

Proof. The proof follows from the fact that any valid bandwidth satisfying coloring of the vertices is a valid coloring of the set of intervals. The reason being that each bandwidth is more than $\frac{1}{2}$. It is now a well known fact that Kierstead and Trotter's online interval coloring algorithm [10] is 3 competitive. Hence the theorem holds. □

3.4. Online strategy and an upper bound for the CIB problem

The colors to be assigned are split into three disjoint classes C_1, C_2, and C_3. These classes are determined in an online fashion. The classes are built dynamically, when a new color is required the first unused color is assigned. When a color is assigned to one of the three classes it can no longer be assigned to any of the other classes.

We perform an online partition of the intervals into 3 disjoint subsequences, S_1, S_2, and S_3 according to their bandwidth. For every interval I:

- $I \in S_1$ if $b(I) \leqslant \frac{1}{4}$;
- $I \in S_2$ if $\frac{1}{4} < b(I) \leqslant \frac{1}{2}$;
- $I \in S_3$ if $b(I) > \frac{1}{2}$.

Algorithm CIB.

Run in parallel the following online algorithms, where $m = 4$.

- A_{small} on S_1 using colors from C_1;
- A_{Middle} on S_2 using colors from C_2;
- A_{KT} on S_3 using colors from C_3;

Theorem 3.7. *Algorithm CIB is* 10-*competitive.*

Completing the analysis: We have split the input sequence into three cases. Each of the three cases has been handled by a separate algorithm operating with a separate set of colors. We have also shown that Kierstead and Trotter's online coloring algorithm can be modified to perform competitively on each case. If the optimum for the three cases are O_1, O_2, and O_3, respectively, then the optimum number of colors used by a bandwidth satisfying coloring is no less than each of the three optima. Using this fact and our analysis of the algorithms for each of the three cases, it follows that the online algorithm outlined above uses at most $3 + \frac{2m}{m-2} + m - 1$ times the optimum number of colors used by a bandwidth satisfying assignment. This expression gives the smallest value for $m = 4$, we conclude that our algorithm is 10-competitive.

This completes the proof of Theorem 3.7.

4. Lower bounds for small bandwidth

Kierstead and Trotter [10] proved a lower bound of 3 for the CI problem. Since CIB is a generalization for CI, this lower bound also applies here. However, what if all the intervals were very small?

In this section we present a lower bound on the competitive ratio of deterministic strategies. In particular, we prove a lower bound of $\frac{4}{3}$ on the competitiveness of any deterministic online strategy, by constructing a sequence of requests of the same bandwidth $b \in [d, e]$ for any $0 \leqslant d, e \leqslant 1$. If d and e are larger than $\frac{1}{2}$ then we can simply use the lower bound of Kierstead and Trotter [10]. So it becomes interesting if d and e are very small.

The remainder of this section is devoted to the lower bound stated in Theorem 4.1.

Theorem 4.1. *There is a lower bound of $\frac{4}{3}$ on the competitive ratio for online coloring of intervals with bandwidth, for any restriction of the interval bandwidths to* $[d, e]$, $0 \leqslant d \leqslant e \leqslant 1$

Proof. We choose a bandwidth $b \in [d, e]$, all the presented intervals in the following construction would have bandwidth b. If there is an n such that $bn = 1$, then the number of colors that the optimal offline algorithm uses equals the maximum total bandwidth of the intersecting intervals in the construction. Note that if there is no integer n and a value b, such that $bn = 1$ then the optimal offline algorithm cannot utilize the maximum capacity of each color. Every color used by the optimal offline algorithm, as well as every color used by any online algorithm, has a maximum total bandwidth of $1 - \delta$ for some $\delta < b$. For simplicity, we assume that $bn = 1$, otherwise the total bandwidth given should be a factor of $1 - \delta$ for some $\delta < b$.

We give the following construction in phases.

Phase 1: Present identical intervals $[0, c]$ with total bandwidth k each of bandwidth b. The first $\frac{2}{3}k$ colors used by the online algorithm will be considered the color class A. Any other color is in the color class denoted by $\bar{A}$. Note that different online algorithms produce different sets of A and $\bar{A}$. The intervals presented in this phase can be colored by k colors by the optimal offline algorithm.

Phase 2: In this phase we present intervals that do not intersect with any intervals of the first phase. I.e., their left endpoint is strictly greater than c. We present the intervals in two sets each with total bandwidth k. We first present the intervals in Set 1, then the intervals in Set 2.

Set 1: Present intervals with total bandwidth k. The left endpoint of all the intervals are larger than c. When intervals are presented we follow this rule: present the interval $[\alpha_{x_1}, \beta_{x_1}]$ with bandwidth b, such that for every interval $[\alpha_{i_1}, \beta_{i_1}]$ previously colored in the color class A by the coloring algorithm, $\alpha_{x_1} < \alpha_{i_1}$ and $\beta_{x_1} > \beta_{i_1}$. Also for every interval $[\alpha_{j_1}, \beta_{j_1}]$ with color in the color class $\bar{A}$, $\alpha_{x_1} > \alpha_{j_1}$, and $\beta_{x_1} < \beta_{j_1}$.

Set 2: Present intervals with a total bandwidth of k. The left endpoints of all the intervals are larger than any of the right endpoint of intervals in Set 1. In this way the intervals of this set do not intersect with any of the intervals of the first set. When intervals are presented in this set we follow a similar but opposite rule: present the interval $[\alpha_{x_2}, \beta_{x_2}]$ with bandwidth b, such that for every interval $[\alpha_{i_2}, \beta_{i_2}]$ previously colored in the color class A by the coloring algorithm, $\alpha_{x_2} > \alpha_{i_2}$ and $\beta_{x_2} < \beta_{i_2}$. Also for every interval $[\alpha_{j_2}, \beta_{j_2}]$ with color in the color class $\bar{A}$, $\alpha_{x_2} < \alpha_{j_2}$, and $\beta_{x_2} > \beta_{j_2}$.

In the first set all the intervals with colors in the color class A are contained in all intervals with color in $\bar{A}$. The second set has the property that all the intervals with color in the color class $\bar{A}$ are contained in all intervals with color in A. For convenience, we arrange the intervals differently to get the structure illustrated in Figs. 1 and 2.

Phase 3: For this phase, we introduce the following notations:

- Let X be the set of all intervals colored by $\bar{A}$ in the first set.
  Let x be the total bandwidth of set X divided by k.
- Let Y be the set of all intervals colored A in the second set.
  Let y be the total bandwidth of set Y divided by k.
- Let $a = \max\{x, y\}$.
- Let r_1 be the right endpoint of the smallest interval in Set 1 colored with a color in A. Note that r_1 is to the left of all the right endpoints of intervals colored by color in A in Set 1.
- Let l_2 be the left endpoint of the smallest interval in Set 2 colored with some color in $\bar{A}$. Note that l_2 is to the right of all the right endpoints of intervals colored by color in $\bar{A}$ in Set 2.

After Phase 2 is completed the adversarial sequence continues. Present $(1 - a)k$ identical intervals $[r_1, l_2]$ such that the optimal offline algorithm can still color the whole construction with only k colors. Those intervals are presented as shown in Fig. 3. Note that in this phase the presented intervals intersect intervals from the previous phase, in particular, all the intervals in X and Y, but do not intersect any of the intervals from Phase 1.

We claim that the online algorithm uses $\frac{4}{3}k$ colors by the end of Phase 2 or Phase 3 by the following two lemmas.

Lemma 4.2. *If the online algorithm uses less than $\frac{4}{3}k$ colors by the end of Phase* 2, *then both x and y are at least $\frac{1}{3}$.*

Proof. The optimal offline algorithm can color the sequence of intervals presented in the three phases with only k colors.

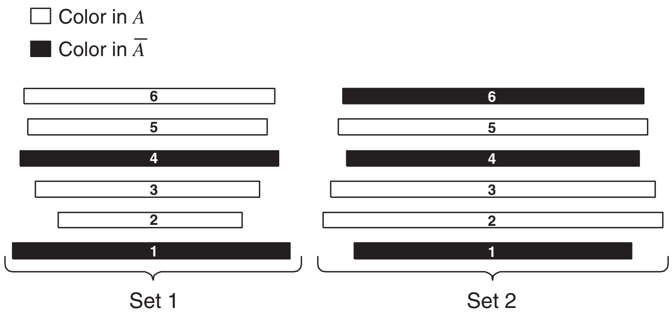

Fig. 1. Intervals colored black are in the color class $\bar{A}$. The white intervals are in the color class A. The numbers inside the intervals indicate the order of their arrival in their set. Note the rule of Set 1 is followed. Each presented interval in this set contains all the previously presented intervals with color in the color class A. Also, each such interval is contained in every previously presented interval with color in the color class $\bar{A}$. Observe that in the second set the opposite rule is followed.

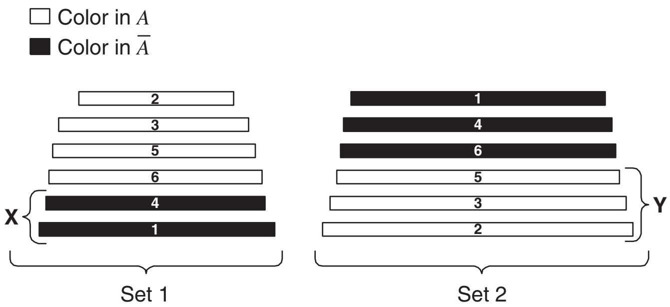

Fig. 2. The two sets after the rearrangement. The intervals are displayed in order of size rather than in order of presentation. In this display, it can be easily seen that intervals colored by color in A in Set 1 are all contained in intervals colored by colors in $\bar{A}$ in Set 1. In Set 2 the opposite is true, every interval colored by some color in A contains all the intervals colored by colors in $\bar{A}$.

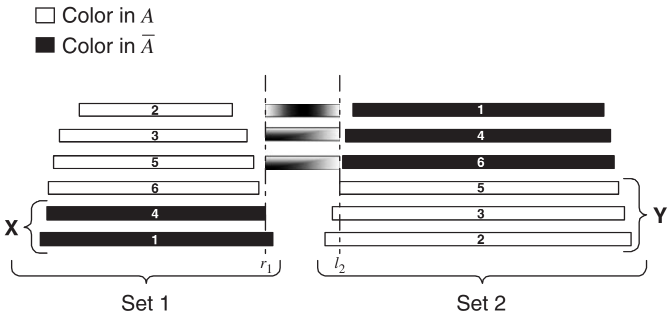

Fig. 3. The final coloring of phases 2 and 3, the identical intervals, colored gray, are placed in the middle of the two sets intersecting all the intervals in Y and X. Note that the total bandwidth of the intervals presented in these phases is still k.

$x \geqslant \frac{1}{3}$: Recall that the color class A has only $\frac{2}{3}k$ colors. Since any online algorithm uses in Set 1 of Phase 2 at least k colors and $(1-x)k$ is at least the number of colors used in A we get $1 - x \leqslant \frac{2}{3} \Rightarrow x \geqslant \frac{1}{3}$.

$y \geqslant \frac{1}{3}$: $1 - y \leqslant \frac{2}{3}$, otherwise the online algorithm uses more than $\frac{2}{3}k$ colors in $\bar{A}$ in Set 1 of Phase 2. In the first phase the online algorithm uses exactly $\frac{2}{3}k$ colors in class A contradicting the assumption that the number of colors used by the online algorithm in Phase 1 and 2 is less than $\frac{4}{3}k$. This gives $y \geqslant \frac{1}{3}$. □

Lemma 4.3. *If x and y are at least $\frac{1}{3}$ then the online algorithm uses more than $\frac{4}{3}k$ colors by the end of Phase* 3.

Proof. Suppose $y \geqslant x$, hence $a = \max\{x, y\} = y$ (the case $a = x$ is similar). Then, the online coloring algorithm uses:

$$((1-y)+y+x)k = (1+x)k \geqslant \left(1+\frac{1}{3}\right)k = \frac{4}{3}k.$$

The first inequality holds due to Lemma 4.2. □

By Lemmas 4.2 and 4.3 the online algorithm uses at least $\frac{4}{3}k$ colors. Since the optimal offline algorithm can use only k colors that implies a lower bound of $\frac{4}{3}$ and completes the proof of Theorem 4.1. □

5. Conclusions and open problems

In this paper we provide a 10-competitive ratio algorithm for the CIB problem, however, the lower bound for the problem remains only 3. There is also a gap for the small bandwidth variant. For this variant we present an asymptotically 2 competitive ratio algorithm and show a lower bound of $\frac{4}{3}$.

References

[1] U. Adamy, T. Erlebach, Online coloring of intervals with bandwidth, in: Proc. First Workshop on Approximation and Online Algorithms, 2003.
[2] Y. Azar, A. Fiat, M. Levy, Improved algorithms for online coloring of intervals with bandwidth, Presented at the Workshop on Online Algorithms 2004, Rungstedgaard, Rungsted Kyst, Denmark, July 2004.
[3] A. Borodin, Ran El-Yaniv, Online Computation and Competitive Analysis, Cambridge University Press, Cambridge, 1998.
[4] M. Chrobak, M. Slusarek, On some packing problems related to dynamic storage allocation, RAIRO Inform. Theoret. Appl. 22 (1988) 487–499.
[5] A. Feldmann, B. Maggs, J. Sgall, D.D. Sleator, A. Tomkins, Competitive analysis of call admission algorithms that allow delay, Technical Report CMU-CS-95-102, Carnegie Mellon University, 1995.
[6] M.C. Golumbic, Algorithmic Graph Theory and Perfect Graphs, Academic Press, New York, 1980.
[7] D.S. Johnson. Near optimal bin packing algorithms, Ph.D. Thesis, MIT, Cambridge, MA, 1973.
[8] H.A. Kierstead, A polynomial approximation algorithm for dynamic storage allocation, Discrete Math. 88 (1991) 231–237.
[9] H.A. Kierstead, J. Qin, Coloring interval graphs with First-Fit, Discrete Math. 144 (1995) 47–57.
[10] H.A. Kierstead, W.T. Trotter Jr., An extremal problem in recursive combinatorics, Congr. Numer. 33 (1981) 143–153.
[11] N.S. Narayanaswamy, Dynamic storage allocation and online colouring interval graphs, in: K.-Y. Chwa, I. Munro (Eds.), Proc. of COCOON 2004, Jeiju Island, South Korea, August 17–20, 2004, Lecture Notes in Computer Science, Vol. 3106, Springer, Berlin, 2004, pp. 274–286.
[12] S.V. Pemmaraju, R. Raman, Improved analysis and algorithms for online coloring of intervals with bandwidth. ⟨http://www.cs.uiowa.edu/∼sriram/papers/bandwidth.pdf⟩.