

Online Coloring of Short Intervals

Joanna Chybowska-Sokół 

Faculty of Mathematics and Information Science, Warsaw University of Technology, Poland
j.sokol@mini.pw.edu.pl

Grzegorz Gutowski 


Institute of Theoretical Computer Science, Faculty of Mathematics and Computer Science,
Jagiellonian University, Kraków, Poland
gutowski@tcs.uj.edu.pl

Konstanty Junosza-Szaniawski 

Faculty of Mathematics and Information Science, Warsaw University of Technology, Poland
k.szaniawski@mini.pw.edu.pl

Patryk Mikos 

Institute of Theoretical Computer Science, Faculty of Mathematics and Computer Science,
Jagiellonian University, Kraków, Poland
mikos@tcs.uj.edu.pl

Adam Polak 

Institute of Theoretical Computer Science, Faculty of Mathematics and Computer Science,
Jagiellonian University, Kraków, Poland
polak@tcs.uj.edu.pl

Abstract

We study the online graph coloring problem restricted to the intersection graphs of intervals with lengths in $[1, \sigma]$. For $\sigma = 1$ it is the class of unit interval graphs, and for $\sigma = \infty$ the class of all interval graphs. Our focus is on intermediary classes. We present a $(1 + \sigma)$ -competitive algorithm, which beats the state of the art for $1 < \sigma < 2$, and proves that the problem we study can be strictly easier than online coloring of general interval graphs. On the lower bound side, we prove that no algorithm is better than $5/3$ -competitive for any $\sigma > 1$, nor better than $7/4$ -competitive for any $\sigma > 2$, and that no algorithm beats the $5/2$ asymptotic competitive ratio for all, arbitrarily large, values of σ . That last result shows that the problem we study can be strictly harder than unit interval coloring. Our main technical contribution is a recursive composition of strategies, which seems essential to prove any lower bound higher than 2.

2012 ACM Subject Classification Theory of computation → Online algorithms

Keywords and phrases Online algorithms, graph coloring, interval graphs

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2020.52

Category APPROX

Related Version A full version of the paper is available at <https://arxiv.org/abs/1802.09503>.

Funding *Joanna Chybowska-Sokół*: Partially supported by the National Science Center of Poland under grant no. 2016/23/N/ST1/03181.

Grzegorz Gutowski: Partially supported by the National Science Center of Poland under grant no. 2016/21/B/ST6/02165.

Patryk Mikos: Partially supported by the National Science Center of Poland under grant no. 2014/14/A/ST6/00138.

Adam Polak: Partially supported by the Polish Ministry of Science and Higher Education program *Diamantowy Grant* under grant no. DI2012 018942.



© Joanna Chybowska-Sokół, Grzegorz Gutowski, Konstanty Junosza-Szaniawski, Patryk Mikos, and Adam Polak;
licensed under Creative Commons License CC-BY

Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2020).

Editors: Jarosław Byrka and Raghu Meka; Article No. 52; pp. 52:1–52:18



Leibniz International Proceedings in Informatics

LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

In the *online graph coloring* problem the input graph is presented to the algorithm vertex by vertex, along with all the edges adjacent to the already presented vertices. Each vertex must be assigned a color, different than any of its neighbors, immediately and irrevocably at the moment it is presented, without any knowledge of the remaining part of the graph. The objective is to minimize the number of used colors. The problem and its variants attract much attention, both for theoretical properties and practical applications in network multiplexing, resource allocation, and job scheduling.

The standard performance measure, used to analyze online algorithms, is the *competitive ratio*, i.e., the worst-case guarantee on the ratio of the solution given by an online algorithm to the optimal offline solution (see Section 1.1 for a formal definition).

In the general case, of online coloring of arbitrary graphs, there is no hope for any algorithm with a constant competitive ratio. The best known algorithm [7] uses $O(\chi \cdot n / \log n)$ colors for n -vertex χ -colorable graphs, i.e. it is $O(n / \log n)$ -competitive, and there is a lower bound [8] showing that no online graph coloring algorithm can be $o(n / \log^2 n)$ -competitive. It is thus common to study the problem restricted to particular graph classes.

Having in mind the applications in scheduling, one of the important special cases is the class of *interval graphs*, i.e. intersection graphs of intervals on the real line. The classic result is by Kierstead and Trotter [11], who designed a 3-competitive algorithm and proved a matching lower bound. However, in the special case of *unit interval graphs*, i.e. intersection graphs of intervals of a fixed length, already the simple greedy FirstFit algorithm is 2-competitive [4].

Coloring unit interval graphs can model only a restricted scheduling setting, with all jobs having the same processing time. On the other hand, allowing arbitrary processing times, modeled by (general) interval graphs, might be a too permissive setting, precluding efficient online algorithms. Thus, we ask what happens in between the interval and unit interval graph classes. In particular, we are interested in the optimal competitive ratio of online coloring algorithms for intersection graphs of intervals of length restricted to a fixed range. Formally, let us introduce the σ -interval coloring problem.

► **Definition 1.** For $\sigma \geq 1$, the σ -interval coloring problem asks: Given a sequence of closed intervals $[l_1, r_1], [l_2, r_2], \dots, [l_n, r_n]$, such that $1 \leq (r_i - l_i) \leq \sigma$ for every $i \in [n]$, find a sequence of colors, c_1, c_2, \dots, c_n , such that

$$\forall_{i \neq j} ([l_i, r_i] \cap [l_j, r_j] \neq \emptyset) \implies (c_i \neq c_j),$$

minimizing the number of distinct colors $|\{c_1, c_2, \dots, c_n\}|$.

We study the problem in the online setting, i.e., intervals are presented one by one, in an arbitrary order, and each interval has to be colored immediately and irrevocably after it is presented.

Note that we choose to include the interval representation in the input, instead of presenting the mere graph. It seems a plausible modeling choice given the scheduling applications. Moreover, it lets algorithms exploit geometric properties of the input, and not only structural graph properties. Naturally, any lower bound obtained for this variant of the problem transfers to the harder variant with no interval representation in the input.

Among others, we look to answer the following two apparent questions.

► **Question 1.** Is σ -interval coloring strictly easier than interval coloring?

► **Question 2.** Is σ -interval coloring strictly harder than unit interval coloring?

Naturally, we ask these questions in the online setting, where easier (harder) mean smaller (greater) best possible competitive ratio.

1.1 Our Results

Before we state our results, let us give a formal definition of the competitive ratio. In this paper we focus on the *asymptotic* competitive ratio.

► **Definition 2.** Let A be an online graph coloring algorithm, and let $A(\chi)$ denote the maximum number of colors A uses to color any graph which can be colored offline using χ -colors (i.e. its chromatic number is at most χ). We say that A has the asymptotic competitive ratio α (or that A is α -competitive, for short), if $\limsup_{\chi \rightarrow \infty} \frac{A(\chi)}{\chi} \leq \alpha$.

Another popular performance measure for online algorithms is the *absolute* competitive ratio, which requires that $\frac{A(\chi)}{\chi} \leq \alpha$ holds for *all* χ (and not only in the limit). The choice of the asymptotic, instead of absolute, competitive ratio for our analysis makes things easier for the algorithm and harder for the lower bounds. In our algorithm, sadly, we do not know how to get rid of a constant additive overhead, which vanishes only with growing χ . This is in contrast to the FirstFit and Kierstead-Trotter algorithms, whose claimed competitive ratios are not only asymptotic but also absolute. The good side is, our lower bounds for the asymptotic competitive ratio imply the identical lower bounds for the absolute competitive ratio.

Algorithm. Our positive result is the existence of a $(1 + \sigma)$ -competitive algorithm.

► **Theorem 3.** For every $\sigma \in \mathbb{Q}$, $\sigma \geq 1$, there is an algorithm for online σ -interval coloring with $1 + \sigma$ asymptotic competitive ratio.

Note that for $\sigma' > \sigma$ every σ' -interval coloring algorithm is also a correct σ -interval coloring algorithm, with the same upper bound on its competitive ratio. Therefore, for $\sigma \in \mathbb{R} \setminus \mathbb{Q}$ Theorem 3 yields an online σ -interval coloring algorithm with a competitive ratio arbitrarily close to $1 + \sigma$. This distinction between rational and irrational values of σ becomes somewhat less peculiar in the light of the results of Fishburn and Graham [5], who proved, among other things, that the classes $C(\sigma)$'s of graphs with interval representation with lengths in $[1, \sigma]$ are *right-continuous*, i.e. $C(\sigma) = \bigcap_{\tau > \sigma} C(\tau)$, exactly at irrational σ 's.

Until now, the state-of-the art was the 2-competitive FirstFit algorithm [4] for $\sigma = 1$, and the 3-competitive Kierstead-Trotter algorithm [11] for $\sigma > 1$. Our algorithm matches the performance of FirstFit for $\sigma = 1$, and beats the Kierstead-Trotter algorithm for $\sigma < 2$.

The algorithm is inspired by the recent result for online coloring of unit disk intersection graphs [9]. We cover the real line with overlapping blocks, grouped into a constant number of classes, with each class constituting a partition of the real line. We assign to each class a private set of available colors. When an interval is presented, the algorithm chooses, in a round-robin fashion, a block containing the interval's left end, and greedily picks a color from the block's class.

While not being overly complicated, the algorithm already answers positively our Question 1. Indeed, since there cannot exist a less-than-3-competitive algorithm for (general) interval coloring [11], σ -interval coloring is strictly easier, for $\sigma < 2$.

Lower Bounds. Our negative results include a series of constructions with the following consequences.

► **Theorem 4.** *For every $\sigma > 1$ there is no online algorithm for σ -interval coloring with the asymptotic competitive ratio less than $5/3$.*

► **Theorem 5.** *For every $\sigma > 2$ there is no online algorithm for σ -interval coloring with the asymptotic competitive ratio less than $7/4$.*

► **Theorem 6.** *For every $\varepsilon > 0$ there is $\sigma \geq 1$ such that there is no online algorithm for σ -interval coloring with the asymptotic competitive ratio $5/2 - \varepsilon$.*

The following, more illustrative, statement is a direct corollary of Theorem 6.

► **Corollary 7.** *There is no online algorithm that works for all $\sigma \geq 1$ and uses at most $2.499 \cdot \omega + f(\sigma)$ colors for ω -colorable graphs (for any function f).*

Theorem 6 gives a positive answer to our Question 2. Indeed, while FirstFit is 2-competitive for unit interval coloring, there is no 2-competitive algorithm for σ -interval coloring (for large enough σ), therefore the latter problem is strictly harder. Working out the exact number from our proof, this starts to be the case for $\sigma > 2^{78}$, however we did not attempt to optimize the constant.

Our proofs of Theorems 4 and 5 can be considered as generalizations of the $3/2$ lower bound for online coloring of unit interval graphs by Epstein and Levy [4]. In particular, we heavily use their *separation strategy*, which also appears in [1].

Our main technical contribution is a recursive composition of strategies, which seems essential to prove any lower bound higher than 2. Our $5/2$ lower bound (Theorem 6) borrows also from the work of Kierstead and Trotter [11]. However, in order to control the length of intervals independently of the number of colors, we cannot simply use the pigeonhole principle, as they did. Instead, we develop Lemmas 19 and 20, which let us overcome this issue, at a cost of a worse bound for the competitive ratio, i.e. $5/2$ instead of 3.

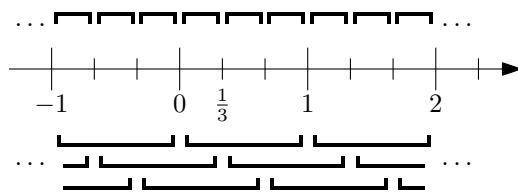
1.2 Related Work

Interval graphs have been intensively studied since the sixties [2, 12], and, in particular, they are known to be *perfect*, i.e. the chromatic number χ of an interval graph always equals the size of the largest clique ω (see, e.g., [6]). To construct an optimal coloring offline it is sufficient to color the graph greedily in a nondecreasing order of the left ends of the intervals.

For the most basic approach for online coloring, that is the FirstFit algorithm, the competitive ratio for interval graphs is still not known exactly. After a series of papers, the most recent results state that FirstFit is at least 5- and at most 8-competitive [10, 13]. Kierstead and Trotter [11] designed a more involved online coloring algorithm, which uses at most $3\omega - 2$ colors for ω -colorable interval graphs, and proved that there exists a strategy that forces any online coloring algorithm to use exactly that number of colors. The same lower and upper bounds were obtained independently by Chrobak and Ślusarek [3, 14]. For intersection graphs of intervals of unit length any online coloring algorithm uses at least $\frac{3}{2}\omega$ colors, and FirstFit uses at most $2\omega - 1$ colors [4].

It seems a natural question to ask if it is possible to improve the bound of $3\omega - 2$ by assuming that interval lengths belong to a fixed range. The study of interval graphs with bounded length representations was initiated by Fishburn and Graham [5]. However, it focused mainly on the combinatorial structure, and not its algorithmic applications.

Kierstead and Trotter [11] give, for every $\omega \in \mathbb{N}_+$, a strategy for Presenter to construct an ω -colorable set of intervals while forcing Algorithm to use at least $3\omega - 2$ colors. However, the lengths of presented intervals increase with the increasing ω . For this reason, with the



■ **Figure 1** Small blocks (above), and large blocks (below), for $b = 3$.

interval length restricted to $[1, \sigma]$, their lower bound is only for the absolute competitive ratio and does not exclude, say, an algorithm that always uses at most $2\omega + \sigma^{10}$ colors. On the contrary, in Theorem 6 we rule out the existence of such an algorithm.

2 Algorithm

► **Theorem 3.** *For every $\sigma \in \mathbb{Q}$, $\sigma \geq 1$, there is an algorithm for online σ -interval coloring with $1 + \sigma$ asymptotic competitive ratio.*

Proof. Let us present an algorithm which, in principle, works for any real σ , however only for a rational σ it achieves the declared competitive ratio. The algorithm has a positive integer parameter b . Increasing the parameter brings the asymptotic competitive ratio closer to $1 + \sigma$ at the cost of increasing the additive constant. More precisely, given an ω -colorable set of intervals our algorithm colors it using at most $\lceil b \cdot (1 + \sigma) \rceil \cdot (\frac{\omega}{b} + b - 1)$ colors, and thus its competitive ratio is $\frac{\lceil b \cdot (1 + \sigma) \rceil}{b} + O(1/\omega)$. For a rational σ , in order to obtain exactly the declared $1 + \sigma$ asymptotic competitive ratio it is sufficient to set b to the smallest possible denominator of a simple fraction representation of σ . Let $\varphi = \lceil b \cdot (1 + \sigma) \rceil$. The algorithm will use colors from the set $\{0, 1, \dots, \varphi - 1\} \times \mathbb{N}$.

Now, let us consider the partition of the real line into *small blocks*. For $i \in \mathbb{Z}$, the i -th small block occupies the interval $[i \cdot \frac{1}{b}, (i + 1) \cdot \frac{1}{b})$. Moreover, we define *large blocks*. The i -th large block occupies the interval $[i \cdot \frac{1}{b}, i \cdot \frac{1}{b} + 1)$. See Figure 1.

Let us point out certain properties of the blocks, which will be useful in the further analysis. Each large block is the union of b consecutive small blocks, and each small block is a subset of b consecutive large blocks. Further, the length of a large block is 1, and for any two intervals of length in $[1, \sigma]$ that both have the left endpoint in the same large block, the two intervals intersect. Thus, the intervals whose left endpoints belong to a fixed large block form a clique. Finally, if the indices of two large blocks differ by at least φ , then any two intervals – one with the left endpoint in one block, the other with the left endpoint in the other – do not intersect.

With each small block the algorithm associates a *small counter*, and with each large block the algorithm associates a *large counter*. Let S_i denote the small counter of the i -th small block, and L_j denote the large counter of the j -th large block. Initially, all the small and large counters are set to zero.

The small counter S_i tracks how many intervals with left endpoint in the i -th small block appeared so far. Based on the small counter the algorithm assigns, in a round-robin fashion, the processed interval to one of the b large blocks containing its left endpoint. The large counter L_j tracks how many intervals were assigned to the j -th large block so far. The color of the processed interval will depend on the large counter.

To assign a color to an interval, the algorithm proceeds as follows:

1. Let i be the index of the small block containing the left endpoint of the interval.
2. Let j be the index of the large block containing the left endpoint of the interval such that $j \equiv S_i \pmod{b}$. Note that there is exactly one such j .

3. Assign to the interval the color $(j \bmod \varphi, L_j)$.
4. Increase the small counter S_i by one.
5. Increase the large counter L_j by one.

First let us argue that the algorithm outputs a proper coloring. Consider any two intervals which were assigned the same color. Let j_1 and j_2 denote the indices of the large blocks selected for these intervals by the algorithm. Since the colors of the two intervals have the same first coordinates, we have that $j_1 \equiv j_2 \pmod{\varphi}$. However, since the second coordinates, which are determined by large counters, are also the same, j_1 and j_2 must be different, and thus they differ by at least φ . Hence the left endpoints of the large blocks j_1 and j_2 are at least $1 + \sigma$ apart, and the two considered intervals do not intersect, thus the coloring is proper.

It remains to bound the number of colors in terms of the clique number ω . Let j be the index of the maximum large counter L_j at the end of the algorithm. Clearly, the algorithm used at most $\varphi \cdot L_j$ colors in total. Let C denote the set of intervals with the left endpoints in the j -th large block and colored with a color in $\{j \bmod \varphi\} \times \mathbb{N}$. Observe that $|C| = L_j$. Let x_k denote the number of intervals in C which have the left endpoint in the k -th small block. Recall that the j -th large block is the union of b small blocks – indexed $j, j + 1, \dots, j + b - 1$ – and thus $L_j = x_j + x_{j+1} + \dots + x_{j+b-1}$. Because of the formula $j \equiv S_i \pmod{b}$ in the step (2) of the algorithm, the large counter L_j is incremented by one out of every b intervals with the left endpoints in any given small block. Hence either $x_k = \lfloor \frac{S_k}{b} \rfloor$ or $x_k = \lfloor \frac{S_k}{b} \rfloor + 1$. In particular

$$S_k \geq b \cdot (x_k - 1) + 1.$$

Let D denote the set of all intervals with the left endpoints in the j -th large block. We can bound the number of intervals in D

$$|D| = \sum_{k=j}^{j+b-1} S_k \geq \sum_{k=j}^{j+b-1} (b \cdot (x_k - 1) + 1) = b \cdot (L_j - b) + b.$$

Recall that D is a clique and thus the clique number ω of the input graph is at least the size of D . Therefore $L_j \leq \frac{\omega + b \cdot (b-1)}{b}$, and the algorithm used at most

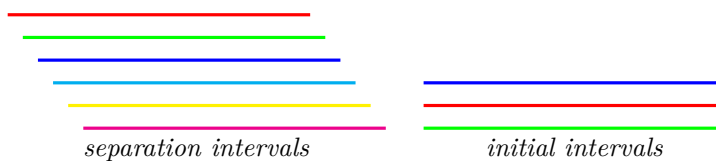
$$\lceil b \cdot (1 + \sigma) \rceil \cdot \left(\frac{\omega}{b} + b - 1 \right)$$

colors. ◀

3 Lower Bounds

3.1 Technical Overview

Algorithm-vs-Presenter game. In order to prove lower bounds for online problems, it is often convenient to look at the problem as a combinatorial game between two players, Algorithm and Presenter. In our case, in each round Presenter reveals an interval, and Algorithm immediately and irrevocably assigns a color to it. Algorithm tries to minimize the number of different colors it assigns. Contrarily, the Presenter’s goal is to force Algorithm to use as many colors as possible, while guaranteeing that the introduced set of intervals is colorable with a smaller number of colors, and contains only short intervals. A strategy for Presenter implies a lower bound on the competitive ratio of any algorithm solving the problem.



■ **Figure 2** Separation strategy.

Separation Strategy. Epstein and Levy [4] prove their $\frac{3}{2}$ lower bound for online coloring of unit intervals by giving the following strategy for Presenter. The strategy has three phases.

1. Presenter introduces a clique of $\frac{\omega}{2}$ equal (i.e. located at the same place) *initial intervals*. Algorithm has to use $\frac{\omega}{2}$ different colors. Let \mathcal{X} denote the set of these colors.
2. Just to the left of the initial intervals, Presenter introduces a clique of ω *separation intervals*. These intervals are not equal, they all have slightly different endpoints, and in total they occupy a place of length $1 + \varepsilon$. None of them intersect the initial intervals. The specific way the separation intervals are presented ensures that all separation intervals with colors in \mathcal{X} have their left (resp. right) endpoints to the left of all left (resp. right) endpoints of separation intervals with colors not in \mathcal{X} (see Figure 2). In particular, each of $\frac{\omega}{2}$ right-most separation intervals has a color not in \mathcal{X} .
3. Presenter introduces a clique of equal $\frac{\omega}{2}$ *final intervals* that intersect all the initial intervals, and $\frac{\omega}{2}$ right-most separation intervals.

Algorithm has to use at least $\frac{3}{2}\omega$ different colors, but the largest clique size (i.e. the offline optimum) is only ω .

The second phase of the above strategy is an example of the *separation strategy*. The formal details are included in the proof of Lemma 14.

Recursive Composition of Strategies. Let us generalize the above strategy. Observe, that instead of presenting a clique in the first phase, Presenter can use an arbitrary strategy, presenting a set of intervals with the clique number equal to $\frac{\omega}{2}$ but possibly enforcing more than $\frac{\omega}{2}$ colors. Moreover, Presenter might be able to achieve a better trade-off by allowing a clique of size $\beta\omega$ (for some $0 < \beta < 1$) in the first phase, and introducing a clique of size $(1 - \beta)\omega$ in the third phase. Actually, it turns out that if the strategy used in the first phase enforces a competitive ratio α (e.g. presenting a clique has $\alpha = 1$), then it is optimal to set $\beta = \frac{1}{1+\alpha}$.

Assume we have a strategy that enforces a competitive ratio α , uses intervals of length at most σ , and needs a place of length M on the real axis (e.g. presenting a clique gives $\alpha = \sigma = M = 1$). Consider the following strategy.

1. Presenter plays the assumed strategy to obtain a set of initial intervals with the clique number $\frac{\omega}{1+\alpha}$. Algorithm has to use $\alpha \frac{\omega}{1+\alpha}$ different colors, denoted by \mathcal{X} .
2. Presenter introduces a clique of ω separation intervals (of unit length), and at least the right-most $\omega - |\mathcal{X}| \geq \frac{\omega}{1+\alpha}$ of them get new colors, not in \mathcal{X} .
3. Presenter introduces a clique of $\alpha \frac{\omega}{1+\alpha}$ final intervals of length $M + \varepsilon$ that intersect all the initial intervals, and the $\frac{\omega}{1+\alpha}$ right-most separation intervals.

In total, the intervals can be colored with ω colors, but the algorithm uses at least

$$\alpha \frac{\omega}{1+\alpha} + \frac{\omega}{1+\alpha} + \alpha \frac{\omega}{1+\alpha} = \frac{(2\alpha + 1)\omega}{1+\alpha}$$

colors, i.e. it is at most $(2 - \frac{1}{\alpha+1})$ -competitive

Let S_0 be a trivial strategy that presents a clique of equal unit intervals. Now, let S_{i+1} be a strategy obtained as above by playing S_i in the first phase. It enforces a competitive ratio α_{i+1} , uses intervals of length at most σ_{i+1} , and needs a place of length M_{i+1} , where $\alpha_0 = \sigma_0 = M_0 = 1$, and each $\alpha_{i+1} = 2 - \frac{1}{\alpha_{i+1}}$, $\sigma_{i+1} = M_i + \varepsilon$, $M_{i+1} = M_i + 1 + \varepsilon$. By solving the recurrence equations we get the following corollary.

► **Corollary 8.** *For every $n \in \mathbb{N}_+$ and every $\varepsilon > 0$, there is no online algorithm for $(n + \varepsilon)$ -interval coloring with the asymptotic competitive ratio less than $\frac{F_{2n+1}}{F_{2n}}$, where F_n is the n -th Fibonacci number ($F_0 = F_1 = 1$, $F_{n+2} = F_{n+1} + F_n$).*

Note that this method cannot give a lower bound higher than $\lim_{n \rightarrow \infty} \frac{F_{2n+1}}{F_{2n}} = \frac{1+\sqrt{5}}{2} \approx 1.61803$. However, we can get arbitrarily close to this bound. That is, for every $\varepsilon > 0$ there is a σ and ω_0 such that for each $\omega \geq \omega_0$ there is a strategy for Presenter to present intervals of length at most σ and force Algorithm to use $\left(\frac{1+\sqrt{5}}{2} - \varepsilon\right) \cdot \omega$ colors on an ω -colorable set of intervals.

► **Observation 9.** *There is no online algorithm that works for all $\sigma \geq 1$ and uses at most $1.618 \cdot \omega + f(\sigma)$ colors for ω -colorable graphs (for any function f).*

Overview of the 5/2 Lower Bound. Our 5/2 lower bound also works by recursively combining strategies, however the three phases are substantially different than above. In particular, in the first phase, instead of playing the simpler strategy only once, Presenter plays many instances of it, independently, side by side. Intuitively, either Algorithm uses mostly different colors for different instances, and it already uses too many colors, or there must be many colors shared by many instances.

So far the argument resembles the lower bound for interval graphs by Kierstead and Trotter [11]. They use pigeonhole principle to argue that if the total number of colors is bounded, then after playing exponentially many instances, there must be four instances that use exactly the same subset of colors. Thus, the number of instances – and consequently the length of intervals in the subsequent phases – required in their approach grows with the number of colors, which makes it infeasible in our setting.

To overcome the above difficulty we show (see Lemmas 19 and 20) that, if the total number of colors is small, there must be four groups of consecutive instances, which do not necessarily use exactly the same colors, but at least have large intersection.

3.2 Preliminaries

To properly capture asymptotic properties of the introduced strategies we give the following formal definitions.

► **Definition 10.** *For $\omega, C \in \mathbb{N}_+$ and $\sigma, M \in \mathbb{R}_+$, an $\langle \omega, C, \sigma, M \rangle$ -strategy is a strategy for Presenter that forces Algorithm to use at least C colors subject to the following constraints:*

1. *the set of introduced intervals is ω -colorable,*
2. *every introduced interval has length at least 1 and at most σ ,*
3. *every introduced interval is contained in the interval $[0, M]$.*

We are interested in providing strategies that achieve the biggest possible ratio $\frac{C}{\omega}$ for large ω . This motivates the following definition.

► **Definition 11.** *An $\langle \alpha, \sigma, M \rangle$ -schema is a set of $\langle \omega, C_\omega, \sigma, M \rangle$ -strategies for all $\omega \in \mathbb{N}_+$ such that $C_\omega = \alpha\omega - o(\omega)$.*

The $o(\omega)$ term in the above definition accounts for the fact that sometimes in a proof we would like to introduce, say, $\frac{\omega}{2}$ -clique. Then, for odd ω 's a rounding is required, which results in small inaccuracies we need to control.

► **Remark 12.** Note that the existence of an $\langle \alpha, \sigma, M \rangle$ -schema implies a lower bound of α for the asymptotic competitive ratio of any online algorithm solving the σ -interval coloring problem.

To put the above definitions in context, note that Kierstead and Trotter [11] give, for all $\omega \in \mathbb{N}_+$, an $\langle \omega, 3\omega - 2, f(\omega), f(\omega) \rangle$ -strategy. However, their family of strategies does not yield an $\langle \alpha, \sigma, M \rangle$ -schema, because the length of the presented intervals grows with ω .

► **Example 13** ($\langle 1, 1, 1 \rangle$ -schema). For any $\omega \in \mathbb{N}_+$, a strategy that introduces the interval $[0, 1]$ in every round $1, \dots, \omega$ is an $\langle \omega, \omega, 1, 1 \rangle$ -strategy. The set of these strategies is a $\langle 1, 1, 1 \rangle$ -schema.

In this section we show a series of constructions that use an existing schema to create another schema with different parameters. The $\langle 1, 1, 1 \rangle$ -schema given above is the initial step for those constructions.

Let S be an $\langle \omega, C, \sigma, M \rangle$ -strategy. We will say that *Presenter uses strategy S in the interval $[x, x + M]$* to denote that Presenter plays according to S , presenting intervals shifted by x , until Algorithm uses C colors.

3.3 Warm-up

Our first construction is a natural generalization of the strategy for unit intervals given by Epstein and Levy [4], already described in Section 3.1 in an informal way. The construction is surpassed by more involved strategies coming later, but it serves as a gentle introduction to our formal framework.

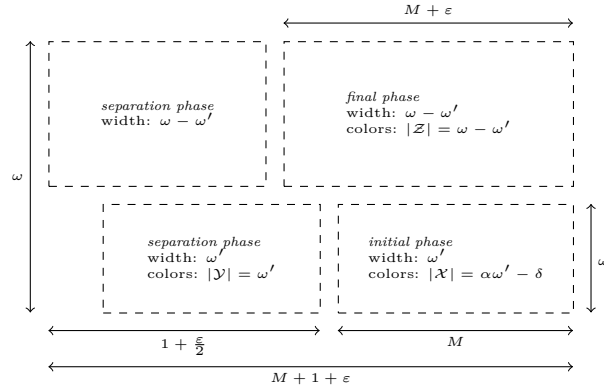
► **Lemma 14.** *If there is an $\langle \alpha, \sigma, M \rangle$ -schema, then there is a $\langle 2 - \frac{1}{\alpha+1}, M + \varepsilon, M + 1 + \varepsilon \rangle$ -schema for every $\varepsilon > 0$.*

Proof. To prove the lemma we need to provide an $\langle \omega, (2 - \frac{1}{\alpha+1})\omega - o(\omega), M + \varepsilon, M + 1 + \varepsilon \rangle$ -strategy for every $\omega \in \mathbb{N}_+$. Let us fix an $\omega \in \mathbb{N}_+$, and let $\omega' = \lfloor \frac{\omega}{\alpha+1} \rfloor$. The $\langle \alpha, \sigma, M \rangle$ -schema contains an $\langle \omega', \alpha\omega' - \delta, \sigma, M \rangle$ -strategy S for some $\delta = o(\omega')$. The strategy for Presenter consists of three phases (see Figure 3). In the first phase, called the *initial phase*, Presenter uses strategy S inside the interval $[1 + \varepsilon, M + 1 + \varepsilon]$. Let $C = \alpha\omega' - \delta$ and let \mathcal{X} denote the set of C colors used by Algorithm in the initial phase.

The second phase, borrowed from [4, 1], is called the *separation phase*. In this phase, Presenter plays the following separation strategy for ω rounds. Let $l_1 = 0$ and $r_1 = \frac{\varepsilon}{2}$. In the i -th round of the separation phase Presenter introduces the interval $[\frac{l_i+r_i}{2}, \frac{l_i+r_i}{2} + 1]$. If Algorithm colors the interval with one of the colors in \mathcal{X} , let $l_{i+1} = \frac{l_i+r_i}{2}$ and $r_{i+1} = r_i$, which means that the next interval will be shifted slightly to the right. Otherwise, let $l_{i+1} = l_i$ and $r_{i+1} = \frac{l_i+r_i}{2}$, which means that the next interval will be shifted slightly to the left.

The above procedure guarantees the following invariant. At the beginning of round i all the previously introduced intervals with a color in \mathcal{X} have their left endpoints to the left of l_i , and, conversely, all the previously introduced intervals with a color not in \mathcal{X} have their left endpoints to the right of r_i . Moreover, all the intervals yet to be introduced will have their left endpoints strictly between l_i and r_i .

Observe that all intervals introduced in the separation phase have length 1 and $\forall_i \frac{l_i+r_i}{2} < \frac{\varepsilon}{2}$. Thus, every interval introduced in the separation phase is contained in $[0, 1 + \frac{\varepsilon}{2}]$ and any two of those intervals intersect. Furthermore, the above invariant guarantees that for any two



■ **Figure 3** Strategy construction in Lemma 14.

intervals x, y introduced in the separation phase, x colored with a color in \mathcal{X} , and y colored with a color not in \mathcal{X} , we have that the left endpoint of x is to the left of the left endpoint of y . Let Y be the set of $\omega' \leq \omega - |\mathcal{X}|$ right-most intervals introduced in the separation phase, and let \mathcal{Y} be the set of colors used by Algorithm on the intervals in Y . Note that \mathcal{X} and \mathcal{Y} are disjoint.

For the last phase, called the *final phase*, let r be the left-most right endpoint of an interval in Y . In the final phase Presenter introduces $\omega - \omega'$ times the same interval $[r, M + 1 + \varepsilon]$. This interval intersects all intervals introduced in the initial phase, all intervals in Y , and no other interval introduced in the separation phase. Thus, Algorithm must use $\omega - \omega'$ colors in the final phase that are different from the colors in both \mathcal{X} and \mathcal{Y} . Let \mathcal{Z} denote the set of colors used by Algorithm in the final phase.

The presented set of intervals is clearly ω -colorable and Algorithm used at least $|\mathcal{X}| + |\mathcal{Y}| + |\mathcal{Z}| = \alpha\omega' - \delta + \omega' + \omega - \omega' = \left(2 - \frac{1}{\alpha+1}\right)\omega - o(\omega)$ many colors. The longest interval presented has length $M + \varepsilon$, and all intervals are contained in $[0, M + 1 + \varepsilon]$. Thus, we have constructed a $\left\langle 2 - \frac{1}{\alpha+1}, M + \varepsilon, M + 1 + \varepsilon \right\rangle$ -schema. ◀

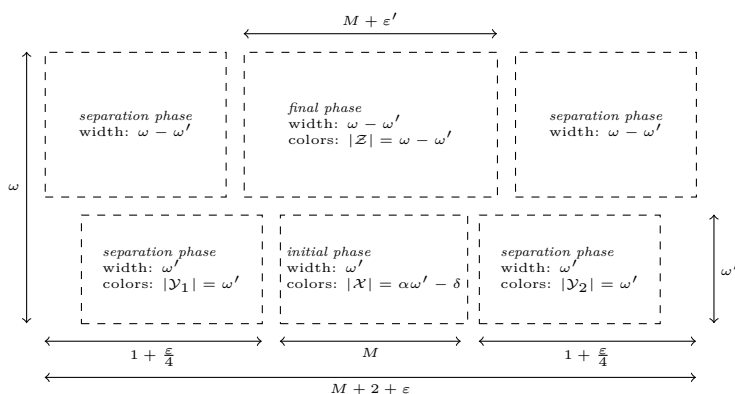
3.4 The 5/3 Lower Bound

► **Lemma 15.** *If there is an $\langle \alpha, \sigma, M \rangle$ -schema, then there is a $\left\langle 2 - \frac{1}{\alpha+2}, M + \varepsilon, M + 2 + \varepsilon \right\rangle$ -schema for every $\varepsilon > 0$.*

Proof. The proof of this lemma is very similar to the proof of Lemma 14, but now we have two separation phases instead of just one, see Figure 4. Let us fix an $\omega \in \mathbb{N}_+$, and let $\omega' = \left\lfloor \frac{\omega}{\alpha+2} \right\rfloor$. Let S be an $\langle \omega', \alpha\omega' - \delta, \sigma, M \rangle$ -strategy for some $\delta = o(\omega')$.

In the initial phase, Presenter uses S inside interval $\left[1 + \frac{\varepsilon}{2}, M + 1 + \frac{\varepsilon}{2}\right]$, and forces Algorithm to use $C = \alpha\omega' - \delta$ colors. Let \mathcal{X} denote the set of those colors.

In the separation phase, Presenter plays the separation strategy two times. First, Presenter plays the separation strategy for ω rounds in the region $\left[0, 1 + \frac{\varepsilon}{4}\right]$ pushing to the right colors not in \mathcal{X} . Let Y_1 be the set of ω' right-most intervals from this first separation. Let \mathcal{Y}_1 denote the set of colors used by Algorithm to color Y_1 . Then, Presenter plays the separation strategy for ω rounds in the region $\left[M + 1 + \frac{3\varepsilon}{4}, M + 2 + \varepsilon\right]$ pushing to the left colors not in $\mathcal{X} \cup \mathcal{Y}_1$. Let Y_2 be the set of ω' left-most intervals from this second separation. Let \mathcal{Y}_2 denote the set of colors used by Algorithm to color Y_2 .



■ **Figure 4** Strategy construction in Lemma 15.

Let r be the left-most right endpoint of an interval in Y_1 . Let l be the right-most left endpoint of an interval in Y_2 . In the final phase Presenter introduces $\omega - \omega'$ times the same interval $[r, l]$.

The presented set of intervals is clearly ω -colorable and Algorithm used at least $|\mathcal{X}| + |\mathcal{Y}_1| + |\mathcal{Y}_2| + |\mathcal{Z}| = \alpha\omega' - \delta + \omega' + \omega' + \omega - \omega' = \left(2 - \frac{1}{\alpha+2}\right)\omega - o(\omega)$ many colors. The longest interval presented has length at most $M + \varepsilon$, and all intervals are contained in $[0, M + 2 + \varepsilon]$. Thus, we have constructed a $\left\langle 2 - \frac{1}{\alpha+2}, M + \varepsilon, M + 2 + \varepsilon \right\rangle$ -schema. ◀

► **Corollary 16.** *There is an $\langle \alpha_n, 2n - 1 + \varepsilon, 2n + 1 + \varepsilon \rangle$ -schema, for every $n \in \mathbb{N}_+$ and every $\varepsilon > 0$, where*

$$\alpha_n = \frac{(\sqrt{3} - 3)(\sqrt{3} - 2)^n + (\sqrt{3} + 3)(-\sqrt{3} - 2)^n}{(\sqrt{3} - 1)(\sqrt{3} - 2)^n + (\sqrt{3} + 1)(-\sqrt{3} - 2)^n}.$$

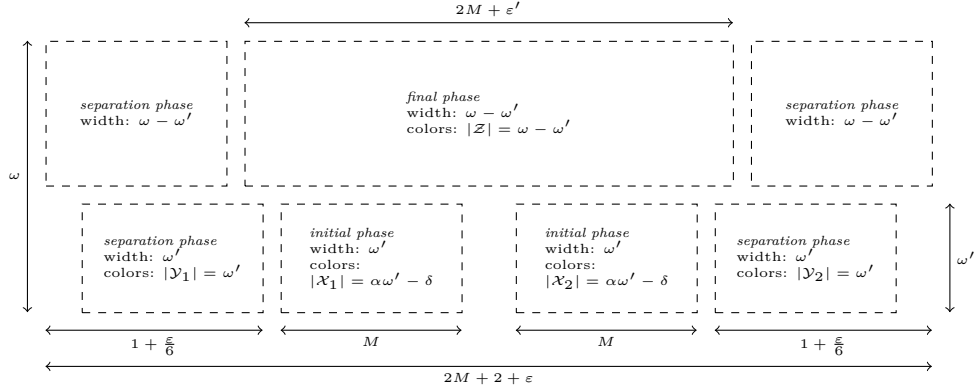
Proof. Starting with a $\langle 1, 1, 1 \rangle$ -schema and repeatedly applying Lemma 15 one can generate¹ a family of $\langle \alpha_n, \sigma_n + \varepsilon_n, M_n + \varepsilon_n \rangle$ -schemas, such that $\alpha_{n+1} = 2 - \frac{1}{\alpha_n+2}$, $\sigma_{n+1} = M_n$, $M_{n+1} = M_n + 2$, and $\alpha_0 = \sigma_0 = M_0 = 1$. Solving the recurrence equations we get $\alpha_n = \frac{F_{2n+1}}{F_{2n}}$, $\sigma_n = 2n - 1$, and $M_n = 2n + 1$. ◀

Note that, similarly to Observation 9, one could already use Corollary 16 to get a lower bound arbitrarily close to $\lim_{n \rightarrow \infty} \alpha_n = \sqrt{3} \approx 1.73205$ for the asymptotic competitive ratio of any online algorithm that work for all $\sigma \geq 1$. Nonetheless in Section 3.6 we prove a stronger $5/2$ lower bound.

► **Theorem 4.** *For every $\sigma > 1$ there is no online algorithm for σ -interval coloring with the asymptotic competitive ratio less than $5/3$.*

Proof. Assume for contradiction that for some $\sigma > 1$ there exists an online algorithm for σ -interval coloring with the asymptotic competitive ratio $\frac{5}{3} - \varepsilon$, for some $\varepsilon > 0$. By the definition of the asymptotic competitive ratio, there is an ω_A such that for every $\omega \geq \omega_A$ the algorithm colors every ω -colorable set of intervals using at most $(\frac{5}{3} - \varepsilon + \frac{\varepsilon}{3})\omega = (\frac{5}{3} - \frac{2\varepsilon}{3})\omega$ colors.

¹ Knowing the desired target values of n and ε , one needs to properly adjust the ε value for each application of Lemma 15, e.g., it is sufficient to set it to ε/n .



■ **Figure 5** Lemma 17, Case 1: $|\mathcal{C}_2 \setminus \mathcal{C}_1| \geq \frac{\omega}{2\alpha+2}$.

Observe that, for $n = 1$, Corollary 16 gives a $\langle \frac{5}{3}, 1 + (\sigma - 1), 3 + (\sigma - 1) \rangle$ -schema. By the definition of schema, there is an ω_P such that for every $\omega \geq \omega_P$ there is a strategy for Presenter to present an ω -colorable set of intervals, of length in $[1, \sigma]$, and force Algorithm to use $(\frac{5}{3} - \frac{\varepsilon}{3})\omega$ colors. For $\omega = \max(\omega_A, \omega_P)$ we reach a contradiction. ◀

3.5 The 7/4 Lower Bound

► **Lemma 17.** *If there is an $\langle \alpha, \sigma, M \rangle$ -schema, then there is a $\langle \frac{2\alpha+1}{2\alpha+2}, 2M + \varepsilon, 2M + 2 + \varepsilon \rangle$ -schema for every $\varepsilon > 0$.*

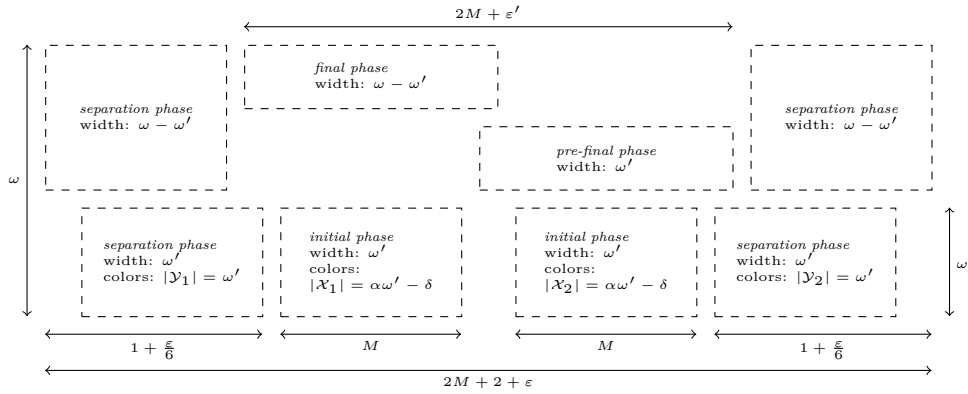
Proof. The proof of this lemma is a bit more complicated than the previous ones, as we now have two initial phases, two separation phases and a strategy branching, see Figure 5 and Figure 6. Let us fix an $\omega \in \mathbb{N}_+$, and let $\omega' = \lfloor \frac{\omega}{\alpha+1} \rfloor$. Let S be an $\langle \omega', \alpha\omega' - \delta, \sigma, M \rangle$ -strategy for some $\delta = o(\omega')$.

In the initial phase, Presenter uses strategy S twice: (1) inside interval $[1 + \frac{\varepsilon}{6}, M + 1 + \frac{\varepsilon}{6}]$, and (2) inside interval $[M + 1 + \frac{2\varepsilon}{3}, 2M + 1 + \frac{2\varepsilon}{3}]$. Algorithm uses $C = \alpha\omega' - \delta$ colors in each of these intervals. We get a set of colors \mathcal{X}_1 used by Algorithm in the first interval, and a set of colors \mathcal{X}_2 used by Algorithm in the second interval. Note that $\mathcal{X}_1 \cap \mathcal{X}_2$ might be non-empty.

In the separation phase, Presenter plays the separation strategy two times. First, Presenter plays the separation strategy for ω rounds in the region $[0, 1 + \frac{\varepsilon}{6}]$ pushing to the right colors not in \mathcal{X}_1 . Let Y_1 be the set of ω' right-most intervals from the first separation phase. Let \mathcal{Y}_1 denote the set of colors used by Algorithm to color Y_1 . Then, Presenter plays the separation strategy for ω rounds in the region $[2M + 1 + \frac{5\varepsilon}{6}, 2M + 2 + \varepsilon]$ pushing to the left colors not in \mathcal{X}_2 . Let Y_2 be the set of ω' left-most intervals from the second separation phase. Let \mathcal{Y}_2 denote the set of colors used by Algorithm to color Y_2 . Let r be the left-most right endpoint of an interval in Y_1 . Let l be the right-most left endpoint of an interval in Y_2 .

There are two cases in the final phase. Let $\mathcal{C}_1 := \mathcal{X}_1 \cup \mathcal{Y}_1$, and analogously $\mathcal{C}_2 := \mathcal{X}_2 \cup \mathcal{Y}_2$. We have that $|\mathcal{C}_1| = |\mathcal{C}_2| = (\alpha + 1)\omega' - \delta = \omega - o(\omega)$.

Case 1. If $|\mathcal{C}_2 \setminus \mathcal{C}_1| \geq \frac{\omega}{2\alpha+2}$, then Presenter introduces $\omega - \omega'$ times the same interval $[r, l]$. Each interval introduced in the final phase intersects with all intervals from both initial phases and all intervals in $Y_1 \cup Y_2$. Thus, Algorithm is forced to use $|\mathcal{C}_1 \cup \mathcal{C}_2| + \omega - \omega' = |\mathcal{C}_1| + |\mathcal{C}_2 \setminus \mathcal{C}_1| + \omega - \omega' \geq \omega - o(\omega) + \frac{\alpha+1}{\alpha+1}\omega = (2 - \frac{1}{2\alpha+2})\omega - o(\omega)$ colors in total.



■ **Figure 6** Lemma 17, Case 2: $|\mathcal{C}_2 \setminus \mathcal{C}_1| < \frac{\omega}{2\alpha+2}$.

Case 2. If $|\mathcal{C}_2 \setminus \mathcal{C}_1| < \frac{\omega}{2\alpha+2}$, then Presenter introduces ω' intervals, all of them having endpoints $[M + 1 + 5\varepsilon/12, l]$. Let Q be the set of colors used by Algorithm in this *pre-final phase*. We have $\mathcal{C}_2 \cap Q = \emptyset$, and we assumed that $|\mathcal{C}_2 \setminus \mathcal{C}_1| \leq \frac{\omega}{2\alpha+2}$, thus we have $|Q \setminus \mathcal{C}_1| \geq \frac{\omega}{2\alpha+2}$, and now we are in a situation analogous to Case 1, with Q playing the role of \mathcal{C}_2 , see Figure 6.

The longest interval introduced by Presenter in both cases has length strictly less than $2M + \varepsilon$, and the whole game is played in the region $[0, 2M + 2 + \varepsilon]$. ◀

► **Corollary 18.** *There is an $\langle \alpha_n, 3 \cdot 2^n - 4 + \varepsilon, 3 \cdot 2^n - 2 + \varepsilon \rangle$ -schema, for every $n \in \mathbb{N}_+$ and every $\varepsilon > 0$, where*

$$\alpha_n = \frac{(\sqrt{7}-4)(\sqrt{7}-3)^n + (\sqrt{7}+4)(-\sqrt{7}-3)^n}{(\sqrt{7}-1)(\sqrt{7}-3)^n + (\sqrt{7}+1)(-\sqrt{7}-3)^n}.$$

Proof. The argument is similar to Corollary 16, but now we solve the recurrence equations $\alpha_0 = 1$, $\alpha_{n+1} = 2 - \frac{1}{2\alpha_n+2}$, and $M_0 = 1$, $M_{n+1} = 2M_n + 2$, $\sigma_0 = 1$, $\sigma_{n+1} = 2M_n$. ◀

Note that, similarly to Observation 9, one could already use Corollary 18 to get a lower bound arbitrarily close to $\lim_{n \rightarrow \infty} \alpha_n = \frac{1+\sqrt{7}}{2} \approx 1.82287$ for the asymptotic competitive ratio of any online algorithm that work for all $\sigma \geq 1$. Nonetheless in Section 3.6 we prove a stronger $5/2$ lower bound.

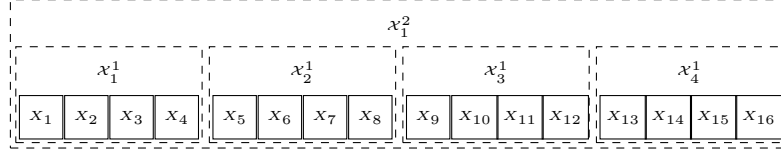
► **Theorem 5.** *For every $\sigma > 2$ there is no online algorithm for σ -interval coloring with the asymptotic competitive ratio less than $7/4$.*

Proof. Observe that, for $\sigma > 2$ and $n = 1$, Corollary 18 gives a $\langle \frac{7}{4}, 2 + (\sigma - 2), 4 + (\sigma - 2) \rangle$ -schema. Then proceed analogously to the proof of Theorem 4. ◀

3.6 The $5/2$ Lower Bound

To prove our main negative result we need two simple combinatorial lemmas.

► **Lemma 19.** *Let $\gamma \in [0, 1]$. For every four sets X_1, \dots, X_4 , each of size k , if their intersection is small: $|\bigcap_{i=1}^4 X_i| \leq (1 - \gamma) \cdot k$, their union is large: $|\bigcup_{i=1}^4 X_i| \geq \frac{3+\gamma}{3} \cdot k$.*



■ **Figure 7** \mathcal{X}_i^j sets in Lemma 20.

Proof. Each element which belongs to the union but does not belong to the intersection can belong to at most three sets. Thus, we have

$$3 \cdot \left(\left| \bigcup_{i=1}^4 X_i \right| - \left| \bigcap_{i=1}^4 X_i \right| \right) \geq 4 \cdot \left(k - \left| \bigcap_{i=1}^4 X_i \right| \right),$$

and so

$$3 \cdot \left| \bigcup_{i=1}^4 X_i \right| \geq 4k - \left| \bigcap_{i=1}^4 X_i \right| \geq (3 + \gamma) \cdot k. \quad \blacktriangleleft$$

► **Lemma 20.** Let $\gamma \in [0, 1]$, and X_1, \dots, X_{4^n} be a family of 4^n sets, each of size k . Then, either

$$\left| \bigcup_{i=1}^{4^n} X_i \right| \geq \left(\frac{3 + \gamma}{3} \right)^n k,$$

or the sequence $1, 2, \dots, 4^n$ can be covered with four disjoint intervals $[l_1, r_1], \dots, [l_4, r_4]$, $l_1 = 1$, $l_{i+1} = r_i + 1$, $r_4 = 4^n$, such that for $Y_i = \bigcup_{j=l_i}^{r_i} X_j$ the intersection of Y_i 's is large:

$$|Y_1 \cap Y_2 \cap Y_3 \cap Y_4| \geq (1 - \gamma) \cdot k.$$

Proof. Consider $n + 1$ families of sets defined as follows: $\mathcal{X}_i^0 := X_i$ for every $i \in [4^n]$, and $\mathcal{X}_i^j := \bigcup_{l=4i-3}^{4i} \mathcal{X}_l^{j-1}$ for every $j \in [n]$ and $i \in [4^{n-j}]$. See Figure 7.

If for some i, j we have $|\bigcap_{l=4i-3}^{4i} \mathcal{X}_l^j| \geq (1 - \gamma) \cdot k$, then we are done. Thus, we assume that $\forall_{i,j} |\bigcap_{l=4i-3}^{4i} \mathcal{X}_l^j| < (1 - \gamma) \cdot k$. Let $\varrho := \frac{3+\gamma}{3} \in [1, \frac{4}{3}]$. We prove that $\forall_{i,j} |\mathcal{X}_i^j| \geq \varrho^j k$, by induction on j . For $j = 0$ the statement is obvious because $\forall_i |\mathcal{X}_i^0| = |X_i| = k = \varrho^0 k$. For $j + 1$ and arbitrary i , let $k' = \varrho^j k$. By the induction hypothesis $|\mathcal{X}_{4i-3}^j|, \dots, |\mathcal{X}_{4i}^j| \geq \varrho^j k = k'$. We may ignore some elements of those sets and assume that $|\mathcal{X}_{4i-3}^j| = \dots = |\mathcal{X}_{4i}^j| = k'$, moreover we assumed that $|\mathcal{X}_{4i-3}^j \cap \dots \cap \mathcal{X}_{4i}^j| < (1 - \gamma)k = \frac{1-\gamma}{\varrho^j} \varrho^j k = (1 - \gamma')k'$, where $\gamma' \in [0, 1]$ and $\gamma' > \gamma$. We apply Lemma 19 and get $|\mathcal{X}_{4i-3}^j \cup \dots \cup \mathcal{X}_{4i}^j| \geq \frac{3+\gamma'}{3} k'$. Thus, $|\mathcal{X}_i^{j+1}| \geq \frac{3+\gamma'}{3} k' > \frac{3+\gamma}{3} k' = \varrho k' = \varrho^{j+1} k$. \blacktriangleleft

► **Lemma 21.** If there is an $\langle \alpha, \sigma, M \rangle$ -schema, then for every $\varepsilon > 0$ and for every $\gamma \in (0, 1)$, there is a $\langle \frac{5}{4} + \frac{1}{2}(1 - \gamma)\alpha, 4^n M + \varepsilon, 4^n M + \varepsilon \rangle$ -schema, for some $n := n(\gamma)$.

Proof. Let us fix an $\omega \in \mathbb{N}_+$, and let $\omega' = \lfloor \frac{\omega}{2} \rfloor$. Let S be an $\langle \omega', \alpha\omega' - \delta, \sigma, M \rangle$ -strategy for some $\delta = o(\omega')$. Presenter repeats strategy S in the initial phase 4^n times. For each $i \in [4^n]$ the i -th game is played inside interval $[(i-1)(M + \frac{\varepsilon}{4^n}), (i-1)(M + \frac{\varepsilon}{4^n}) + M]$. See Figure 8. Algorithm uses $C = \alpha\omega' - \delta$ colors in each of these games. Let \mathcal{X}_i denote the set of C colors used by Algorithm in the i -th game. Let \mathcal{X} denote the set of all colors used in the initial phase, i.e., $\mathcal{X} = \bigcup_{i \in [4^n]} \mathcal{X}_i$.

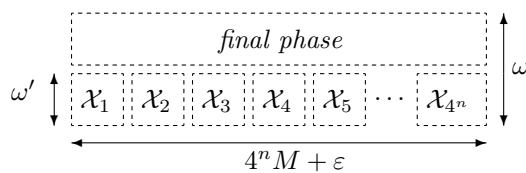


Figure 8 Case 1: $|\mathcal{X}|$ is large.

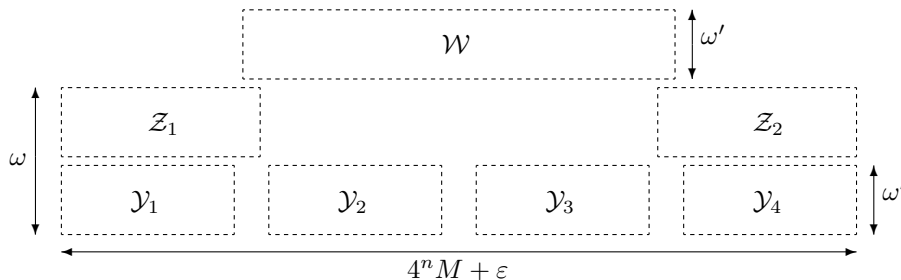


Figure 9 Case 2.1: $|\mathcal{Y}|$ is large and $|\mathcal{Z}_2 \setminus \mathcal{Z}_1| \geq \frac{1}{4}\omega$.

We apply Lemma 20 to the family $\mathcal{X}_1, \dots, \mathcal{X}_{4^n}$ and get that either the union of these sets has at least $\left(\frac{3+\gamma}{3}\right)^n C$ elements, or we get four disjoint consecutive subfamilies $\mathcal{Y}_1, \dots, \mathcal{Y}_4$ ($\mathcal{Y}_i = \bigcup_{j=i}^{4^n} \mathcal{X}_j$) such that the size of the intersection $\mathcal{Y}_1 \cap \mathcal{Y}_2 \cap \mathcal{Y}_3 \cap \mathcal{Y}_4$ has at least $(1-\gamma) \cdot C$ elements.

Case 1. If the size of the union $|\mathcal{X}|$ is at least $(1 + \frac{\gamma}{3})^n \cdot C$, then Presenter introduces ω' intervals, all of them having endpoints $[0, 4^n M + \varepsilon]$. See Figure 8. Each interval introduced in the final phase intersects with all intervals introduced in the initial phase. Thus, Algorithm is forced to use at least $|\mathcal{X}| + \omega' \geq \frac{1}{2} \left((1 + \frac{\gamma}{3})^n \alpha + 1 \right) \omega - o(\omega)$ colors in total. Easy calculation shows that for $\gamma \in (0, 1)$, $\alpha \in [1, 3]$ and for any $n \geq \log_{1+\frac{\gamma}{3}}(5/2 - \gamma)$, we have $\frac{1}{2} + \frac{1}{2} \left(1 + \frac{\gamma}{3} \right)^n \alpha \geq \frac{5}{4} + \frac{1}{2}(1 - \gamma)\alpha$.

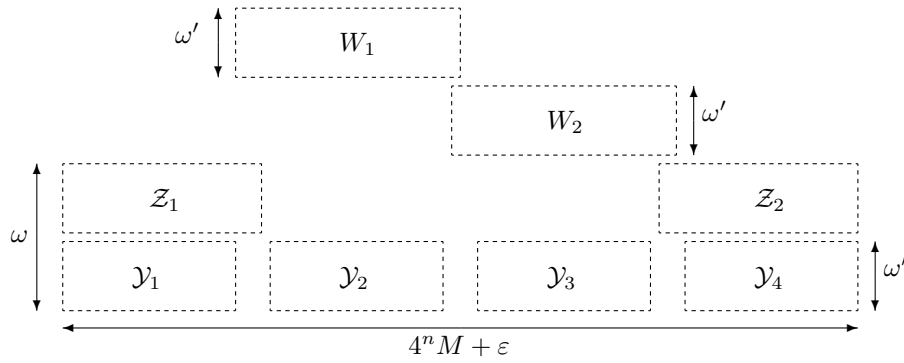
Case 2. The size of the intersection $|\mathcal{Y}_1 \cap \dots \cap \mathcal{Y}_4|$ is at least $(1 - \gamma) \cdot C$. Let $\mathcal{Y} = \mathcal{Y}_1 \cap \mathcal{Y}_2 \cap \mathcal{Y}_3 \cap \mathcal{Y}_4$ denote the colors that appear in all four parts of the initial phase. Presenter introduces a set \mathcal{Z}_1 of ω' identical intervals covering all intervals contributing to \mathcal{Y}_1 and disjoint with intervals contributing to \mathcal{Y}_2 . See Figure 9. Let \mathcal{Z}_1 be the set of colors used by Algorithm to color \mathcal{Z}_1 .

Then Presenter introduces a set \mathcal{Z}_2 of ω' identical intervals covering all intervals contributing to \mathcal{Y}_4 and disjoint with intervals contributing to \mathcal{Y}_3 . Let \mathcal{Z}_2 be the set of colors used by Algorithm to color \mathcal{Z}_2 .

Clearly, $|\mathcal{Z}_1| = |\mathcal{Z}_2| = \omega'$, and $\mathcal{Z}_1 \cap \mathcal{Y} = \mathcal{Z}_2 \cap \mathcal{Y} = \emptyset$. Now we distinguish two subcases depending on the size of the set $\mathcal{Z}_2 \setminus \mathcal{Z}_1$.

Case 2.1. If $|\mathcal{Z}_2 \setminus \mathcal{Z}_1| \geq \frac{1}{4}\omega$, then Presenter introduces a set \mathcal{W} of ω' identical intervals intersecting all the intervals in \mathcal{Z}_1 and \mathcal{Z}_2 , and covering all the intervals contributing to \mathcal{Y}_2 and \mathcal{Y}_3 . Let \mathcal{W} be the set of colors used by Algorithm to color \mathcal{W} . By the definition, we have $\mathcal{W} \cap \mathcal{Y} = \mathcal{W} \cap \mathcal{Z}_1 = \mathcal{W} \cap \mathcal{Z}_2 = \emptyset$. Algorithm was forced to use $|\mathcal{W}| + |\mathcal{Z}_1| + |\mathcal{Z}_2 \setminus \mathcal{Z}_1| + |\mathcal{Y}| \geq \left(\frac{1}{2} + \frac{1}{2} + \frac{1}{4} \right) \omega + \frac{1}{2}(1 - \gamma)\alpha\omega - o(\omega) = \left(\frac{5}{4} + \frac{1}{2}(1 - \gamma)\alpha \right) \omega - o(\omega)$ colors in total. See Figure 9.

Case 2.2. If $|\mathcal{Z}_2 \setminus \mathcal{Z}_1| < \frac{1}{4}\omega$, then let $\mathcal{Z} = \mathcal{Z}_1 \cap \mathcal{Z}_2$ and observe that $|\mathcal{Z}| \geq \lfloor \frac{\omega}{4} \rfloor$. Presenter introduces a set \mathcal{W}_1 of ω' identical intervals intersecting all the intervals in \mathcal{Z}_1 , and covering all the intervals contributing to \mathcal{Y}_2 . Then, presenter introduces a set \mathcal{W}_2 of



■ **Figure 10** Case 2.2: $|\mathcal{Y}|$ is large and $|\mathcal{Z}_2 \setminus \mathcal{Z}_1| < \frac{1}{4}\omega$.

ω' identical intervals, intersecting all the intervals in W_1 and Z_2 , and covering all the intervals contributing to \mathcal{Y}_3 . Let \mathcal{W} be the set of colors used by Algorithm to color intervals in $W_1 \cup W_2$. We have that $|\mathcal{W}| = 2\omega'$, and $\mathcal{W} \cap \mathcal{Y} = \mathcal{W} \cap \mathcal{Z} = \emptyset$. Algorithm was forced to use $|\mathcal{W}| + |\mathcal{Z}| + |\mathcal{Y}| \geq (1 + \frac{1}{4})\omega + \frac{1}{2}(1 - \gamma)\alpha\omega - o(\omega) = (\frac{5}{4} + \frac{1}{2}(1 - \gamma)\alpha)\omega - o(\omega)$ colors in total. See Figure 10. ◀

► **Corollary 22.** *There is an $\langle \alpha_n, 4^{nf(\gamma)} + \varepsilon, 4^{nf(\gamma)} + \varepsilon \rangle$ -schema, for every $n \in \mathbb{N}_+$, every $\varepsilon > 0$, and every $\gamma \in (0, 1)$, where*

$$\alpha_n = \frac{5}{2} \cdot \frac{1}{1 + \gamma} - \frac{(3 - 2\gamma)}{2(1 + \gamma)} \cdot \left(\frac{1 - \gamma}{2}\right)^n, \quad f(\gamma) = \left\lceil \frac{\log(\frac{5}{2} - \gamma)}{\log(1 + \frac{\gamma}{3})} \right\rceil.$$

Proof. The argument is similar to Corollaries 16, and 18, but now we solve the recurrence equations $\alpha_0 = 1$, $\alpha_{n+1} = \frac{5}{4} + \frac{1}{2}(1 - \gamma)\alpha_n$ for competitive ratio, and $M_0 = 1$, $M_{n+1} = 4^{f(\gamma)}M_n$, $\sigma_n = M_n$ for region and interval lengths. ◀

► **Theorem 6.** *For every $\varepsilon > 0$ there is $\sigma \geq 1$ such that there is no online algorithm for σ -interval coloring with the asymptotic competitive ratio $5/2 - \varepsilon$.*

Proof. Assume for contradiction that, for some $\varepsilon > 0$, there are $(5/2 - \varepsilon)$ -competitive algorithms for every $\sigma \geq 1$. Setting γ small enough and n large enough, Corollary 22 gives us a $\langle \frac{5}{2} - \frac{\varepsilon}{4}, \sigma, \sigma \rangle$ -schema, for some value of σ . This means, there is ω_P such that for every $\omega \geq \omega_P$ there exists an $\langle \omega, (\frac{5}{2} - \frac{2\varepsilon}{4})\omega, \sigma, \sigma \rangle$ -strategy. On the other hand, for the assumed σ -interval coloring algorithm, there exists ω_A such that for every $\omega \geq \omega_A$ the algorithm uses at most $(\frac{5}{2} - \frac{3\varepsilon}{4})\omega$ colors for every ω -colorable set of intervals. For $\omega = \max(\omega_A, \omega_P)$ we reach a contradiction. ◀

References

- 1 Yossi Azar, Amos Fiat, Meital Levy, and N.S. Narayanaswamy. An improved algorithm for online coloring of intervals with bandwidth. *Theoretical Computer Science*, 363(1):18–27, 2006. doi:10.1016/j.tcs.2006.06.014.
- 2 Seymour Benzer. On the topology of the genetic fine structure. *Proceedings of the National Academy of Sciences of the United States of America*, 45(11):1607–1620, 1959. doi:10.1073/pnas.45.11.1607.
- 3 Marek Chrobak and Maciej Ślusarek. On some packing problem related to dynamic storage allocation. *RAIRO, Theoretical Informatics and Applications*, 22(4):487–499, 1988. URL: http://www.numdam.org/item/ITA_1988__22_4_487_0.

- 4 Leah Epstein and Meital Levy. Online interval coloring and variants. In *ICALP 2005: 32nd International Colloquium on Automata, Languages and Programming, Lisbon, Portugal, July 2005. Proceedings*, volume 3580 of *Lecture Notes in Computer Science*, pages 602–613, 2005. doi:10.1007/11523468_49.
- 5 P. C. Fishburn and R. L. Graham. Classes of interval graphs under expanding length restrictions. *Journal of Graph Theory*, 9(4):459–472, 1985. doi:10.1002/jgt.3190090405.
- 6 Martin Charles Golumbic. *Algorithmic Graph Theory and Perfect Graphs (Annals of Discrete Mathematics, Vol 57)*. Elsevier, 2 edition, 2004.
- 7 Magnús M. Halldórsson. Parallel and on-line graph coloring. *Journal of Algorithms*, 23(2):265–280, 1997. doi:10.1006/jagm.1996.0836.
- 8 Magnús M. Halldórsson and Mario Szegedy. Lower bounds for on-line graph coloring. *Theoretical Computer Science*, 130(1):163–174, 1994. doi:10.1016/0304-3975(94)90157-0.
- 9 Konstanty Junosza-Szaniawski, Paweł Rzażewski, Joanna Sokół, and Krzysztof Węsek. Online coloring and L(2,1)-labeling of unit disk intersection graphs. *SIAM Journal on Discrete Mathematics*, 32(2):1335–1350, 2018. doi:10.1137/16M1097821.
- 10 Henry A. Kierstead, David A. Smith, and William T. Trotter. First-fit coloring on interval graphs has performance ratio at least 5. *European Journal of Combinatorics*, 51:236–254, 2016. doi:10.1016/j.ejc.2015.05.015.
- 11 Henry A. Kierstead and William T. Trotter. An extremal problem in recursive combinatorics. In *12th Southeastern Conference on Combinatorics, Graph Theory and Computing, Baton Rouge, LA, USA, March 1981. Proceedings, vol. II*, volume 33 of *Congressus Numerantium*, pages 143–153, 1981. URL: <https://people.math.gatech.edu/~trotter/papers/29.pdf>.
- 12 C. Lekkeikerker and J. Boland. Representation of a finite graph by a set of intervals on the real line. *Fundamenta Mathematicae*, 51(1):45–64, 1962. URL: <http://eudml.org/doc/213681>.
- 13 N. S. Narayanaswamy and R. Subhash Babu. A note on first-fit coloring of interval graphs. *Order*, 25(1):49–53, 2008. doi:10.1007/s11083-008-9076-6.
- 14 Maciej Ślusarek. A coloring algorithm for interval graphs. In *MFCS 1989: Symposium on Mathematical Foundations of Computer Science, Porąbka-Kozubnik, Poland, August 1989. Proceedings*, volume 379 of *Lecture Notes in Computer Science*, pages 471–480, 1989. doi:10.1007/3-540-51486-4_93.

A Open Problems

Throughout the paper we presented several constructions which can be combined recursively to obtain strategies proving higher and higher lower bounds for larger and larger interval lengths σ . Table 1 summarizes a selection of these strategies.

■ **Table 1** Summary of selected strategies for Presenter.

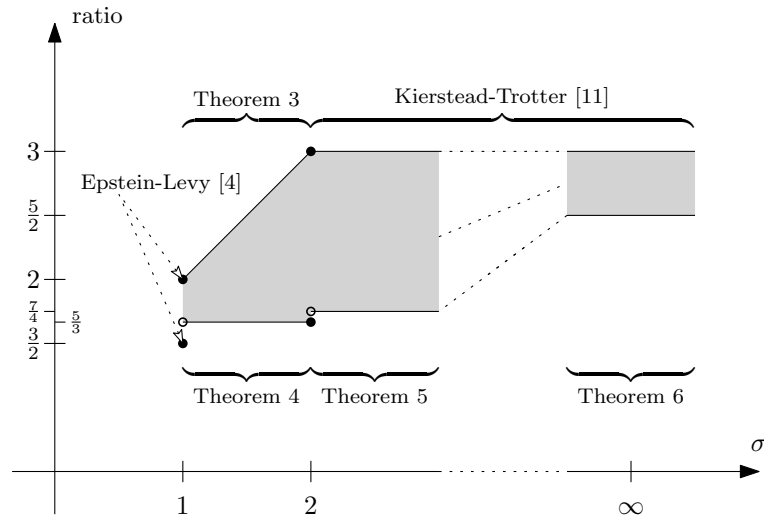
ratio	interval length	strategy
1.5	1	Epstein and Levy [4]
1.6	$2 + \varepsilon$	Corollary 8, $n = 2$ iterations
1.66	$1 + \varepsilon$	Corollary 16, $n = 1$ iteration
1.72	$3 + \varepsilon$	Corollary 16, $n = 2$ iterations
1.75	$2 + \varepsilon$	Corollary 18, $n = 1$ iteration
1.81	$8 + \varepsilon$	Corollary 18, $n = 2$ iterations
2	$4^{39} + \varepsilon$	Corollary 22, $n = 3$ iterations, $\gamma = 0.21030395$
2.4	$4^{486} + \varepsilon$	Corollary 22, $n = 6$ iterations, $\gamma = 0.0339$
2.49	$4^{7970} + \varepsilon$	Corollary 22, $n = 10$ iterations, $\gamma = 0.003449$

52:18 Online Coloring of Short Intervals

There are still large gaps between the best known lower and upper bounds for the optimal competitive ratios for online σ -interval coloring problems (see Figure 11). It would be interesting to close the gap, even for a single specific σ . For example, for $\sigma = 3/2$ the optimal online algorithm has the competitive ratio somewhere between $5/3$ and $5/2$.

Finally, let us conjecture that the lower bound of Theorem 6 is tight.

► **Conjecture 23.** *There is a $5/2$ -competitive online algorithm for σ -interval coloring, for every $\sigma \geq 1$.*



■ **Figure 11** Gap between current bounds for competitive ratio of online σ -interval coloring.