

Comparing First-Fit and Next-Fit for Online Edge Coloring

Martin R. Ehmsen^{1,*}, Lene M. Favrholt^{1,*}, Jens S. Kohrt^{1,*},
and Rodica Mihai²

¹ Department of Mathematics and Computer Science,
University of Southern Denmark
{ehmsen, lenem, svalle}@imada.sdu.dk

² Department of Informatics, University of Bergen
rodica.mihai@ii.uib.no

Abstract. We study the performance of the algorithms First-Fit and Next-Fit for two online edge coloring problems. In the min-coloring problem, all edges must be colored using as few colors as possible. In the max-coloring problem, a fixed number of colors is given, and as many edges as possible should be colored. Previous analysis using the competitive ratio has not separated the performance of First-Fit and Next-Fit, but intuition suggests that First-Fit should be better than Next-Fit. We compare First-Fit and Next-Fit using the relative worst order ratio, and show that First-Fit is better than Next-Fit for the min-coloring problem. For the max-coloring problem, we show that First-Fit and Next-Fit are not strictly comparable, i.e., there are graphs for which First-Fit is better than Next-Fit and graphs where Next-Fit is slightly better than First-Fit.

1 Introduction

In edge coloring, the edges of a graph must be colored such that no two adjacent edges receive the same color. This paper studies two variants of online edge coloring, min-coloring and max-coloring. For both problems, the algorithm is given the edges of a graph one by one, each one specified by its endpoints.

In the *min-coloring* problem, each edge must be colored before the next edge is received, and once an edge has been colored, its color cannot be changed. The aim is to color all edges *using as few colors as possible*.

For the *max-coloring* problem, a limited number k of colors is given. Each edge must be either colored or rejected before the next edge arrives. Once an edge has been colored, its color cannot be changed and it cannot be rejected. Similarly, once an edge has been rejected, it cannot be colored. In this problem, the aim is to *color as many edges as possible*.

For both problems we study the following two algorithms. *First-Fit* is the natural greedy algorithm which colors each edge using the lowest possible color.

* Supported in part by the Danish Agency for Science, Technology and Innovation (FNU).

Next-Fit uses the colors in a cyclic order. It colors the first edge with the color 1 and keeps track of the last used color c_{last} . For the max-coloring problem, when coloring an edge (u, v) , it uses the first color in the sequence $\langle c_{\text{last}} + 1, c_{\text{last}} + 2, \dots, k, 1, 2, \dots, c_{\text{last}} \rangle$ that is not yet used on any edge incident to u or v . For the min-coloring problem it only cycles through the set of colors that it has been forced to use so far.

Both algorithms are members of more general families of algorithms. For the max-coloring problem, we define the class of *fair* algorithms that never reject an edge, unless all k colors are already represented at adjacent edges. For the min-coloring problem, we define the class of *parsimonious* algorithms that do not take a new color into use, unless necessary.

The min-problem has previously been studied in [1], where the main result implies that all parsimonious algorithms have the same competitive ratio of approximately 2.

The max-problem was studied in [8]. For k -colorable graphs, First-Fit and Next-Fit have very similar competitive ratios of $1/2$ and $k/(2k - 1)$. For general graphs, there is an upper bound on the competitive ratio of First-Fit of $\frac{2}{9}(\sqrt{10} - 1) \approx 0.48$, and the competitive ratio of Next-Fit exactly matches the general lower bound for fair algorithms of $2\sqrt{3} - 3 \approx 0.46$. No fair algorithm can be better than 0.5-competitive, so the competitive ratio cannot vary much between fair algorithms. Moreover, there is a general upper bound (even for randomized algorithms) of $4/7 \approx 0.57$.

General intuition suggests that First-Fit should be better than Next-Fit, and thus comes the motivation to study the performance of the two algorithms using some other measure than the competitive ratio. There are previous problems, such as paging [5,3], bin packing [4], scheduling [7], and seat reservation [6] where the relative worst-order ratio was successfully applied and separated algorithms that the competitive ratio could not. The relative worst-order ratio is a quality measure that compares two online algorithms directly, without an indirect comparison via an optimal offline algorithm. Thus, the relative worst-order ratio in many cases give more detailed information than the competitive ratio.

For the min-problem, we prove that the two algorithms are comparable, and First-Fit is 1.5 times better than Next-Fit. For the max-problem, surprisingly, we conclude that First-Fit and Next-Fit are not comparable using the relative worst-order ratio.

2 Quality Measures

The standard quality measure for online algorithms is the competitive ratio. Roughly speaking, the competitive ratio of an online algorithm A is the worst-case ratio of the performance of A to the performance of an optimal offline algorithm over all possible request sequences [14,10].

In this paper, we use the competitive ratio only for the min-coloring problem. For that problem, the measure is defined in the following way. Let A be

an edge coloring algorithm and let E be a sequence of edges. Then, $A(E)$ denotes the number of colors used by A . OPT denotes an optimal offline. The competitive ratio of algorithm A is

$$\text{CR}_A = \inf\{c \mid \exists b: \forall E: A(E) \leq c \cdot \text{OPT}(E) + b\}$$

The relative worst-order ratio was first introduced in [4] in an effort to combine the desirable properties of the max/max ratio [2] and the random-order ratio [11]. The measure was later refined in [5]. We describe the measure using the terminology of the coloring problems. Let E be a sequence of n edges. If σ is a permutation on n elements, then $\sigma(E)$ denotes E permuted by σ .

For the max-coloring problem, $A(E)$ is the number of edges colored by algorithm A , and

$$A_W(E) = \min_{\sigma} \{A(\sigma(E))\}.$$

For the min-coloring problem, $A(E)$ is the number of colors used by A , and

$$A_W(E) = \max_{\sigma} \{A(\sigma(E))\}.$$

Thus, in both cases, $A_W(E)$ is the performance of A on a worst possible permutation of E .

Definition 1. For any pair of algorithms A and B , we define

$$\begin{aligned} c_u(A, B) &= \inf\{c \mid \exists b: \forall E: A_W(E) \leq cB_W(E) + b\} \text{ and} \\ c_l(A, B) &= \sup\{c \mid \exists b: \forall E: A_W(E) \geq cB_W(E) - b\}. \end{aligned}$$

If $c_l(A, B) \geq 1$ or $c_u(A, B) \leq 1$, the algorithms are said to be comparable and the relative worst-order ratio $\text{WR}_{A,B}$ of algorithm A to algorithm B is defined. Otherwise, $\text{WR}_{A,B}$ is undefined.

$$\begin{aligned} \text{If } c_u(A, B) \leq 1, \text{ then } \text{WR}_{A,B} &= c_l(A, B), \text{ and} \\ \text{if } c_l(A, B) \geq 1, \text{ then } \text{WR}_{A,B} &= c_u(A, B). \end{aligned}$$

Intuitively, c_l and c_u can be thought of as tight lower and upper bounds, respectively, on the performance of A relative to B .

3 Min-coloring Problem

We first study the min-coloring problem, where all edges of a graph must be colored using as few colors as possible. The first result is an immediate consequence of a result in [1].

Theorem 1. Any parsimonious algorithm has a competitive ratio of $2 - 1/\Delta$, where Δ is the maximum vertex degree.

Proof. In [1], it is proven that, for any online algorithm A, there is a graph G with maximum vertex degree Δ , such that G can be Δ -colored, but A uses $2\Delta - 1$ colors. On the other hand, since no edge is adjacent to more than $2\Delta - 2$ other edges, no parsimonious algorithm will use more than $2\Delta - 1$ colors. \square

Thus, the competitive ratio does not distinguish First-Fit and Next-Fit. However, with the relative worst-order ratio, we get the result that First-Fit is better than Next-Fit:

Theorem 2. *The relative worst-order ratio of Next-Fit to First-Fit is at least $\frac{3}{2}$.*

The theorem follows directly from Lemmas 1 and 2 below.

Lemma 1. *Given any graph G with edges E , $NF_W(E) \geq FF_W(E)$.*

Proof. For any First-Fit coloring, we construct an ordering of the edges so that Next-Fit does the same coloring as First-Fit. Assume that First-Fit uses k colors and let C_i denote the set of edges that First-Fit colors with color i . The ordering of the edges given to Next-Fit consists of all the edges from C_1 , then from C_2 and further till C_k . The edges in each set is given in an arbitrary order. By the First-Fit policy, each edge in C_i is adjacent to edges of C_1, \dots, C_{i-1} . Thus, since Next-Fit only cycles through the colors that it used so far, it will color the edges the same way as First-Fit. This means that, for any First-Fit coloring, we can construct an ordering of the edges such that Next-Fit uses the same number of colors. The result follows. \square

Lemma 2. *There exists a graph with edges E such that $NF_W(E) \geq \frac{3}{2} FF_W(E)$.*

Proof. For any even k , consider the graph S consisting of $k/2 + 2$ stars $S_0, S_1, \dots, S_{k/2}$ and S_{center} as depicted in Figure 1. The center vertices of $S_1, \dots, S_{k/2}$ have

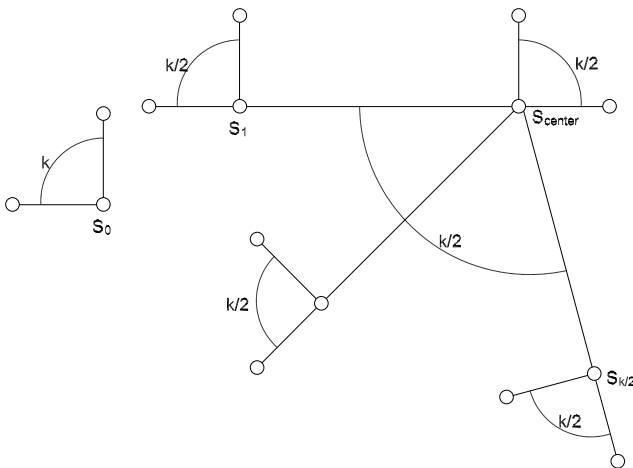


Fig. 1. The graph S used in the proof of Lemma 2

degree $k/2 + 1$, and the centers of S_0 and S_{center} have degree k . Edges for which one end-vertex has degree one are called outer edges. The remaining edges are called inner edges.

It is not difficult to see that, for any ordering of the edges, First-Fit uses exactly k colors: At most one outer edge in each star S_i , $i = 1, \dots, k/2$, is colored with a color larger than $k/2$, and if this happens, the edge connecting S_i to S_{center} has already been colored.

Next-Fit will use $3k/2$ colors, if the edges are given in the following order: First the edges of S_0 , forcing Next-Fit to use the first k colors. Then the outer edges of S_1 followed by the outer edges of S_{center} . Then the inner edge of S_1 , which is colored with the color $k + 1$. Finally, for $i = 2, \dots, k/2$, the outer edges of S_i followed by the inner edge of S_i , which is colored with the color $k + i$. This way, the inner edges will be colored with $k + 1, \dots, 3k/2$. \square

4 Max-coloring Problem

In this section, we study the max-coloring problem, where a limited number k of colors are given, and as many edges as possible should be colored. We first describe a bipartite graph with edges E , such that $\text{FF}_W(E) \geq 9/8 \cdot \text{NF}_W(E)$. Then, we describe a family of graphs with edge set E_n such that $\text{NF}_W(E_n) = (1 + \Omega(\frac{1}{k^2})) \cdot \text{FF}_W(E_n)$. Thus, the two algorithms are not comparable.

4.1 First-Fit Can Be Better Than Next-Fit

Let $B_{k,k} = (X, Y, E)$ be a complete bipartite graph with $|X| = |Y| = k$. For simplicity, we assume that 4 divides k . For other values of k , we get similar results, but the calculations are a bit more messy. We denote by C_i the edges that First-Fit colors with color i .

Proposition 1. *In any First-Fit coloring of $B_{k,k}$, $|C_i| \geq k - i + 1$, $i = 1, \dots, k$.*

Proof. Assume that color i has been used $j \leq k - i$ times. The induced subgraph containing all vertices *not* adjacent to an edge colored with color i is the complete bipartite graph $B_{k-j, k-j}$, where $k - j \geq i$. This subgraph cannot be colored with the colors $1, \dots, i - 1$ only, and since this is a First-Fit coloring, the color i is going to be used. Thus, at least one more edge will be colored with color i . \square

Proposition 2. *If First-Fit colors at most $\frac{9}{16}k^2$ edges of $B_{k,k}$, then*

$$|C_i| \geq \frac{7k^2}{16(2k - 1 - i)}, \quad i = 1, \dots, k.$$

Proof. If First-Fit colors at most $9k^2/16$ edges, then it rejects at least $7k^2/16$ edges. Each rejected edge is adjacent to at least one edge of each color $i = 1, \dots, k$. Each edge colored with color i has $2k - 2$ neighbor edges. Among those, at least $i - 1$ edges are already colored, since each edge colored with i is adjacent

to all lower colors $1, 2, \dots, i - 1$. Thus, at most $2k - 1 - i$ edges can be rejected for each edge colored with i . Hence, for First-Fit to reject $7k^2/16$ edges, it has to use color i at least $7k^2/(16(2k - 1 - i))$ times. \square

Lemma 3. *Given any ordering of the edges of $B_{k,k}$, First-Fit colors more than $\frac{9}{16}k^2$ edges.*

Proof. The number of edges colored by First-Fit is $\sum_{i=1}^k |C_i|$. We assume for the sake of contradiction that First-Fit colors at most $9k^2/16$ edges of $B_{k,k}$. Using Propositions 1 and 2 we get,

$$\begin{aligned} \sum_{i=1}^k |C_i| &\geq \sum_{i=1}^{3k/4} (k - i + 1) + \sum_{i=3k/4+1}^k \frac{7k^2}{16(2k - 1 - i)} \\ &= \sum_{i=k/4+1}^k i + \frac{7}{16}k^2 \sum_{i=k-1}^{5k/4-2} \frac{1}{i} \\ &> \frac{15}{32}k^2 + \frac{7}{16}k^2 \ln\left(\frac{k + k/4 - 2}{k - 2}\right) > \frac{15}{32}k^2 + \frac{7}{16}k^2 \ln(1 + 1/4) \\ &> \frac{15}{32}k^2 + \frac{7}{16}k^2 \frac{3}{14} = \frac{9}{16}k^2, \end{aligned}$$

which is a contradiction. Thus, First-Fit colors more than $\frac{9k^2}{16}$ edges. \square

Lemma 4. *Given the worst ordering of the edges of $B_{k,k}$, Next-Fit colors at most $k^2/2$ edges.*

Proof. We partition the vertex sets X, Y into equal-sized sets X_1, X_2, Y_1, Y_2 . The induced subgraphs H_1 and H_2 with vertex sets X_1, Y_1 and X_2, Y_2 , respectively, are complete bipartite graphs. We give the edges of H_1 and H_2 alternately such that Next-Fit colors the edges of H_1 using colors $1, 2, \dots, k/2$ and the edges of H_2 with colors $k/2 + 1, \dots, k$. After that, Next-Fit cannot color any of the $k^2/2$ edges between H_1 and H_2 . Thus, Next-Fit colors at most $k^2/2$ edges of $B_{k,k}$. \square

Combining Lemmas 3 and 4, we arrive at:

Corollary 1. *Given the graph $B_{k,k} = (X, Y, E)$, $\text{FF}_W(E) \geq \frac{9}{8} \text{NF}_W(E)$.*

4.2 Next-Fit Can Be Slightly Better Than First-Fit

In this section, we prove that there exists a family of graphs where Next-Fit is $1 + \Omega(\frac{1}{k^2})$ times better than First-Fit. We first define the building blocks of the graph family.

Definition 2. *For any given number $k \geq 25$ of colors, a superstar S_k is a graph consisting of an inner star with k edges, each incident to the center of an outer star with $k - 2$ edges of its own.*

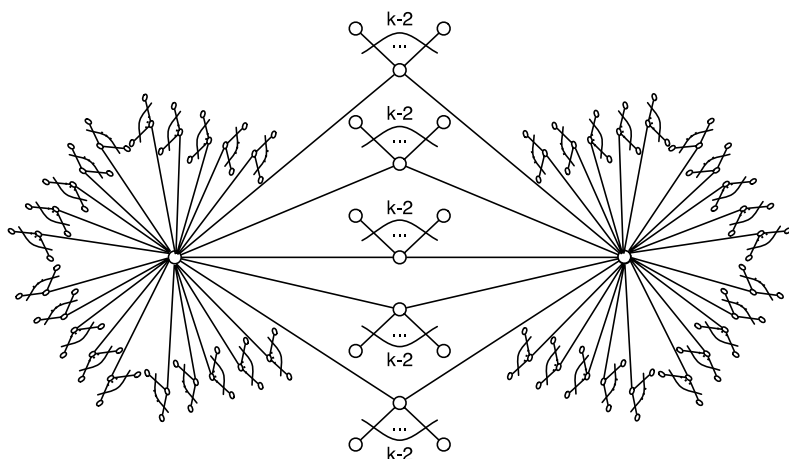


Fig. 2. Two superstars, for $k = 25$, connected through a link of five outer stars

A superstar graph is a graph consisting of superstars. Each pair of superstars in the graph may share a number of outer stars. The set of outer stars shared by a pair of superstars is called the link between them. All outer stars are contained in a link. Each link contains at least five outer stars, and each superstar has links to between five and seven other superstars. See Figure 2 for an incomplete example.

Clearly, fair algorithms never reject outer star edges. However, if all outer stars are colored using the same $k - 2$ colors, at least $k - 2$ edges of each inner star will be rejected. This leads to the following lemma.

Lemma 5. Let $G_{n,k}$ be a superstar graph with n superstars. Then, on its worst ordering of the edges, First-Fit rejects at least $n(k - 2)$ edges.

What remains to be shown is that there exists a family $G_{n,k}$ of superstar graphs, such that on a worst ordering of the edges of $G_{n,k}$, Next-Fit rejects only $n(k - 2) - \Omega(n)$ edges.

Proposition 3. Consider a superstar graph G colored by a fair algorithm. Any superstar in G has at most $k - 1$ edges rejected. If some superstar S in G has $k - 1$ edges rejected, then each of its neighbor superstars has at most $k - 4$ edges rejected.

Proof. Clearly, outer star edges are not rejected, so we only need to consider the inner star edges. At least one inner star edge will be colored in each superstar, since each inner star edge is only adjacent to $k - 1$ edges that are not inner star edges in the same superstar. Thus, at most $k - 1$ edges are rejected from any superstar in the graph.

Assume that some superstar S has $k - 1$ inner star edges rejected. Each of these edges must be adjacent to k colored edges. However, at most $k - 1$ of

these colored neighbor edges belong to S ($k - 2$ from the outer star, and the one colored inner edge of S). Hence, the k th colored neighbor edge must be an inner star edge in a neighboring superstar. Since each link contains at least five inner edges of S and at most one of them is colored, this completes the proof. \square

By Proposition 3, any pair of neighboring superstars have at most $2k - 4$ rejected edges in total. A pair of neighboring superstars with $2k - 4$ rejected edges in total is called a *bad pair*. Note that in a bad pair, exactly $k - 2$ edges are rejected in each superstar. A pair of neighboring superstars with at most $2k - 5$ rejected edges in total is called a *good pair*. A superstar contained only in bad pairs is called a *bad superstar*. A superstar contained in at least one good pair is called a *good superstar*.

Counting the good superstars, the extra colored edge from a good pair is counted at most eight times: once for the superstar S containing it and once for each of the at most seven neighbors of S . Thus, the following lemma follows directly from Proposition 3.

Lemma 6. *Consider a fair coloring of a superstar graph with n superstars. If there are m good superstars, then at most $n(k - 2) - \frac{m}{8}$ edges are rejected.*

Thus, we just need to show that we can connect our building blocks, the superstars, such that, for any Next-Fit coloring, there will be $\Omega(n)$ good superstars. Such a construction is described in the proof of Lemma 8. The proof of Lemma 8 uses Proposition 4 and Lemma 7 below.

The *majority coloring* of a superstar is the set of colors used on the majority of its outer stars, breaking ties arbitrarily. An outer star is *isolated*, if it is not adjacent to at least one colored inner star edge.

Proposition 4. *If two neighboring superstars have different majority colorings, one of them is a good superstar.*

Proof. We prove the proposition by contraposition. Assume that two superstars S and S' are both bad superstars. Then, by Proposition 3, S , S' , and their neighbors each have exactly $k - 2$ edges rejected. Let c_1 and c_2 be the two colors used on inner star edges in S .

If S has m neighbors, the outer stars of S are adjacent to at most $2m + 2$ colored inner star edges. Thus S has at least $k - 2m - 2$ isolated outer stars. Each of these outer stars must be colored with the $k - 2$ colors different from c_1 and c_2 . Hence, the isolated outer stars in S are colored the same, and since $m \geq 5$ and $k \geq 5m$ that coloring is the majority coloring of S . The same is true for S' . Since S and S' have exactly two colored edges and there are at least five edges in the link between them, they share at least one isolated outer star. This means that S and S' have the same majority coloring. \square

Lemma 7. *Assume that $k \geq 101$. Consider a Next-Fit coloring of a superstar graph $G_{n,k}$, $n \geq 6$. Among the bad superstars, there are at most $\frac{2}{3}n$ superstars with the same majority coloring.*

Proof. Any subgraph of $G_{n,k}$ containing x superstars has at least $x\frac{k}{2}$ outer stars. Thus, in any subgraph H of $G_{n,k}$ consisting of x bad superstars with the same majority coloring \mathcal{M} , there are at least $x\frac{k-16}{2} = x(\frac{k}{2} - 8)$ isolated outer stars colored with \mathcal{M} . Each time Next-Fit has used the colors in \mathcal{M} once, the two colors $c_1, c_2 \notin \mathcal{M}$ must be used once, before it will use the colors in \mathcal{M} on isolated outer stars again. Thus, an upper bound on the number of times c_1 and c_2 are used in $G_{n,k}$ gives an upper bound on x .

Clearly, c_1 and c_2 are each used at most once on inner star edges in each superstar. Inside H , c_1 and c_2 are not used on isolated outer stars. Thus, since each bad superstar has at least $k - 16$ isolated outer stars, c_1 and c_2 are used at most $17x$ times on superstars in H .

Outside H , c_1 and c_2 can each be used at most once per outer star, since using c_1 (c_2) on an inner star edge would prohibit the algorithm from using c_1 (c_2) on the adjacent outer star. Hence, since each superstar outside H share each outer star with another superstar, the superstars outside H can only contribute $(n - x)\frac{k}{2}$.

Thus, to create x bad superstars with majority coloring \mathcal{M} , we must have

$$x\left(\frac{k}{2} - 8\right) - 1 \leq 17x + (n - x)\frac{k}{2}.$$

Solving for x , we obtain $x \leq \frac{2}{3}n$, since $k \geq 101$ and $n \geq 6$. \square

Lemma 8. *For $k \geq 101$, there exists a family of superstar graphs $G_{n,k}$ where any Next-Fit coloring results in $\Omega(n)$ good superstars.*

Proof. We use a result from expander graphs [12,9]. Using the notation from [13], for any positive integer m , there exists an $(n = 2m^2, 7, \frac{2-\sqrt{3}}{2})$ -expander, i.e., a 7-regular bipartite multigraph $G(X \cup Y, E)$ with $|X| = |Y| = \frac{n}{2}$, such that for any $S \subseteq X$,

$$|\Gamma(S)| \geq \left(1 + \frac{2 - \sqrt{3}}{2} \left(1 - \frac{2|S|}{n}\right)\right) |S|,$$

where $\Gamma(S)$ is the set of edges between S and \bar{S} . The result also holds for any $S \subseteq Y$. The graph contains parallel edges, but each vertex has at least five neighbors. Replacing each set of parallel edges by one edge, we obtain a simple graph with the same Γ -function.

Now, we connect the superstars as in the simple expander graph. For any suitable n , let each vertex in the expander graph correspond to a superstar. Each edge in the expander graph corresponds to a link between the corresponding superstars. Thus, we obtain a superstar graph where each superstar has links to 5 to 7 other superstars.

Consider any Next-Fit coloring of this graph with n superstars. If there are at least $\frac{1}{3}n$ good superstars, the result follows immediately. Thus, we consider the case where there are at least $\frac{2}{3}n$ bad superstars. By Lemma 7, no majority coloring occurs on more than $\frac{2}{3}n$ bad superstars. Among the bad superstars, let S be the superstars with the most frequently occurring majority coloring. If

$|S| < \frac{1}{3}n$, add the bad superstars with the most frequently occurring majority coloring among the superstars not in S . Continue doing this until S reaches a size between $\frac{1}{3}n$ and $\frac{2}{3}n$. This is possible, since we consider the case where there are at least $\frac{2}{3}n$ bad superstars.

Define $S_X = S \cap X$ (similarly for S_Y), and assume without loss of generality that $|S_X| \geq |S_Y|$. Note that $|S_X| \geq \frac{1}{2}|S| \geq \frac{1}{6}n$. We can bound the size of $\Gamma(S)$ from below by the following

$$\begin{aligned} |\Gamma(S)| &\geq |\Gamma(S_X)| - |S_Y| \\ &\geq \frac{2 - \sqrt{3}}{2} \left(1 - \frac{2|S_X|}{n}\right) |S_X| + (|S_X| - |S_Y|). \end{aligned} \tag{1}$$

We now have two cases depending on the size of S_X .

- $\frac{5}{12}n \leq |S_X| \leq \frac{1}{2}n$. Since $|S_X| + |S_Y| \leq \frac{2}{3}n$, we must have $|S_Y| \leq \frac{3}{12}n$. Thus, inequality (1) immediately yields a lower bound of $\frac{5}{12}n - \frac{3}{12}n = \frac{1}{6}n$, since $\frac{2-\sqrt{3}}{2} \left(1 - \frac{2|S_X|}{n}\right)$ is nonnegative.
- $|S_X| < \frac{5}{12}n$. Since $|S_X| - |S_Y| \geq 0$, inequality (1) gives a lower bound of $\left(\frac{2-\sqrt{3}}{2}\right) \frac{1}{6}|S_X| \geq \frac{2-\sqrt{3}}{144}n$.

Hence, in the coloring done by Next-Fit, we in both cases have $\Omega(n)$ links between S and \bar{S} . By the construction of S , each superstar in \bar{S} linked to a superstar s in S is a good superstar or has a different majority coloring than s . Thus, by Proposition 4, there are $\Omega(n)$ good superstars. \square

This immediately yields the following theorem.

Theorem 3. *First-Fit and Next-Fit are not comparable by the relative worst-order ratio.*

Proof. By Lemma 5, there is an ordering of the edges in any superstar graph with n superstars, such that First-Fit rejects at least $n(k - 2)$ edges.

By Lemmas 6 and 8, there are superstar graphs $G_{n,k}$ with n superstars such that, for any ordering of the edges, Next-Fit rejects only $n(k - 2) - \Omega(n)$ edges. Hence, since First-Fit colors $\Theta(nk^2)$ edges,

$$\text{NF}_W(G_{n,k}) = \left(1 + \Omega\left(\frac{1}{k^2}\right)\right) \text{FF}_W(G_{n,k}).$$

On the other hand, by Corollary 1, there exists a graph S , such that

$$\text{FF}_W(S) \geq \frac{9}{8} \text{NF}_W(S). \quad \square$$

5 Conclusion

We have proven that, with the relative worst-order ratio, First-Fit is strictly better than Next-Fit for the min-coloring problem. This is in contrast to the

competitive ratio which is the same for all parsimonious algorithms, a class of algorithms to which First-Fit and Next-Fit belongs.

For the max-coloring problem, the answer is not as clear: With the relative worst-order ratio, there are graphs where First-Fit does significantly better than Next-Fit and graphs where Next-Fit does slightly better than First-Fit. This is somewhat in keeping with an earlier result saying that the two algorithms can hardly be distinguished by their competitive ratios.

Note that, for the max-coloring problem, the two algorithms may be *asymptotically* comparable [5]. Roughly speaking, it means that, as k tends to infinity, the algorithms “become comparable”. This is left as an open problem. Note that if one were to prove that the algorithms are *not* asymptotically comparable, another construction than the superstar graphs would be required: even if Next-Fit colored all edges of a superstar graph, it would color only $1 + \Theta(\frac{1}{k})$ times as many edges as First-Fit.

References

1. Bar-Noy, A., Motwani, R., Naor, J.: The greedy algorithm is optimal for on-line edge coloring. *Information Processing Letters* 44(5), 251–253 (1992)
2. Ben-David, S., Borodin, A.: A new measure for the study of on-line algorithms. *Algorithmica* 11(1), 73–91 (1994)
3. Boyar, J., Ehmsen, M.R., Larsen, K.S.: Theoretical evidence for the superiority of LRU-2 over LRU for the paging problem. In: *Approximation and Online Algorithms*, pp. 95–107 (2006)
4. Boyar, J., Favrholt, L.M.: The relative worst order ratio for on-line algorithms. *ACM Transactions on Algorithms* 3(22) (2007)
5. Boyar, J., Favrholt, L.M., Larsen, K.S.: The relative worst-order ratio applied to paging. *Journal of Computer and System Sciences* 73, 818–843 (2007)
6. Boyar, J., Medvedev, P.: The relative worst order ratio applied to seat reservation. *ACM Transactions on Algorithms* 4(4), article 48, 22 pages (2008)
7. Epstein, L., Favrholt, L.M., Kohrt, J.: Separating online scheduling algorithms with the relative worst order ratio. *Journal of Combinatorial Optimization* 12(4), 362–385 (2006)
8. Favrholt, L.M., Nielsen, M.N.: On-line edge coloring with a fixed number of colors. *Algorithmica* 35(2), 176–191 (2003)
9. Gabber, O., Galil, Z.: Explicit constructions of linear-sized superconcentrators. *Journal of Computer and System Sciences* 22, 407–420 (1981)
10. Karlin, A.R., Manasse, M.S., Rudolph, L., Sleator, D.D.: Competitive snoopy caching. *Algorithmica* 3, 79–119 (1988)
11. Kenyon, C.: Best-fit bin-packing with random order. In: *7th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 359–364 (1996)
12. Margulis, G.A.: Explicit constructions of concentrators. *Problems of Information Transmission* 9(4), 325–332 (1973)
13. Motwani, R., Raghavan, P.: *Randomized Algorithms*. Cambridge University Press, Cambridge (1995)
14. Sleator, D.D., Tarjan, R.E.: Amortized efficiency of list update and paging rules. *Communications of the ACM* 28(2), 202–208 (1985)