# A new algorithm for on-line coloring bipartite graphs [*]

H. J. Broersma [1]

A. Capponi [2]

D. Paulusma [1]

[1] *Department of Computer Science*
*Durham University*
*South Road, Durham, DH1 3LE, U.K.*
`{hajo.broersma,daniel.paulusma}@durham.ac.uk`

[2] *Computer Science, Division of Engineering and Applied Sciences*
*California Institute of Technology, U.S.A.*
`acapponi@cs.caltech.edu`

June 7, 2007

## Abstract

We first show that for any bipartite graph $H$ with at most five vertices, there exists an on-line competitive algorithm for the class of $H$-free bipartite graphs. We then analyze the performance of an on-line algorithm for coloring bipartite graphs on various subfamilies. The algorithm yields new upper bounds for the on-line chromatic number of bipartite graphs. We prove that the algorithm is on-line competitive for $P_7$-free bipartite graphs, i.e., that do not contain an induced path on seven vertices. The number of colors used by the on-line algorithm for $P_6$-free and $P_7$-free bipartite graphs is bounded by roughly twice, respectively roughly eight times the on-line chromatic number. In contrast, it is known that there exists no competitive on-line algorithm to color $P_6$-free (or $P_7$-free) bipartite graphs, i.e., for which the number of colors is bounded by any function only depending on the chromatic number.

**Key words.** on-line coloring, bipartite graph, (on-line) competitive

**AMS subject classifications.** 05C15, 05C85

## 1  Introduction

In static optimization problems one is often faced with the challenge of determining efficient algorithms that solve a particular problem (nearly) optimally for any given instance of the problem. This task is usually facilitated if the structure of the instances is pretty straightforward. As an example, it is a trivial exercise to determine an algorithm for finding a 2-coloring of a given bipartite graph.

---

[*] An extended abstract of some of the results in this paper was presented at the 6th Italian conference on Algorithms and Complexity 2006 [3].

1

In the area of dynamic optimization the situation gets more complicated. There, one often lacks the knowledge of the complete instances of the problems. As an illustration, compare the previous problem with the slightly changed situation in which the bipartite graph is presented on-line, i.e., vertex by vertex, and the algorithm has to assign a color irrevocably to a vertex as it comes in, i.e., only based on the knowledge of the subgraph that has been revealed so far. This slight change of the problem formulation makes it a lot more difficult: whereas the static problem was trivial, no algorithm for the dynamic problem can guarantee an optimal solution for every instance. In [12] it has been shown that the worst-case performance ratio between on-line and off-line coloring of a known input graph on $n$ vertices is at least $2n/(\log_2 n)^2$. It is even questionable whether one can expect to determine an on-line algorithm that does reasonably well, in the sense that the number of colors used is bounded in some other reasonable way, e.g., as a function of some invariant of the input instances.

In this paper we will focus on particular questions of this type related to coloring bipartite graphs. This type of questions in a more general setting is at the heart of the areas of on-line algorithms and of approximation algorithms.

We first give a short historical excursion starting with a benchmark paper due to Gyárfás and Lehel [9]. They introduced the concept of on-line coloring as a general approach. This was motivated by their translation of a rectangle packing problem related to dynamical storage allocation appearing in [4] into an on-line coloring problem. The latter problem was to decide whether the on-line coloring algorithm known as *First-Fit (FF)* has a constant worst-case performance ratio on the family of interval graphs. We note that since [9] many papers on on-line (coloring) problems have appeared. We refer to [14] for a survey on on-line coloring and to [2] for more background on the general area of on-line algorithms.

In order to have some measure of the performance of on-line algorithms, the notion of competitive algorithms has been introduced in [20] and specifically for coloring in [9]. Intuitively, an on-line coloring algorithm is said to be competitive for a family of graphs $\mathcal{G}$, if for any graph $G \in \mathcal{G}$, the number of colors used by the algorithm on $G$ is bounded from above by a function only depending on the chromatic number of $G$. The chromatic number of $G$ is the smallest number of colors that is necessary to properly color the vertices of $G$ (off-line), i.e., such that adjacent vertices receive different colors. In [13] it is shown that *FF* is competitive for interval graphs, with a bounding function that is linear in the chromatic number, and in [6] competitiveness of *FF* for geometric intersection graphs has been proven. In [5] it is shown that *FF* is competitive for graphs with a bounded independence number, including co-planar graphs.

It is well-known that *FF* is not competitive for $P_6$-free bipartite graphs, i.e., bipartite graphs that do not contain an induced path on six vertices: if the vertices of a complete bipartite graph $K_{m,m}$ minus a perfect matching $\{u_1, v_1\}, \{u_2, v_2\}, \ldots, \{u_m, v_m\}$ are presented in the ordering $u_1, v_1, u_2, v_2, \ldots, u_m, v_m$, then *FF* uses $m$ colors. In fact, there are many families of graphs for which it has been proved that no competitive algorithms exist: two examples given in [9] are the family of trees and the family of $P_6$-free bipartite graphs. These negative results have led to the definition of a weaker form of competitiveness in [7], namely on-line competitiveness, although results of this type have been obtained before the term was formally introduced.

We can explain the notion of on-line competitiveness as follows. It is perhaps more natural to compare the number of colors that is needed to color a graph $G$ by an on-line algorithm to the on-line chromatic number instead of the chromatic number. We will define the on-line chromatic number formally in Section 3. Intuitively, it is the number of colors used by the best performing

on-line algorithm for $G$, i.e., that gives the smallest number of colors in the worst-case ordering of the presented vertices of $G$. An on-line coloring algorithm is said to be on-line competitive if the number of colors is bounded from above by a function only depending on the on-line chromatic number of $G$. The main open problem is whether on-line competitive coloring algorithms exist for all graphs. A subproblem is to establish such algorithms for certain classes of graphs for which competitive algorithms do not exist. Solving this subproblem can be particularly useful if somebody must design an on-line coloring algorithm in the case that the input graphs are only known to be in a specified class of graphs. This also motivated our choice for considering special graph classes.

It is shown in [10] that $FF$ is on-line competitive for trees; it is even optimal for trees, in the sense that if $FF$ uses $k$ colors, then the on-line chromatic number of the tree is also $k$. In [7] it is shown that $FF$ is on-line competitive with an exponential bounding function for graphs with girth at least five.

In the context of algorithmic graph theory it has become rather fashionable to consider forbidden subgraph conditions. For instance, many NP-hard problems turn out to be solvable in polynomial time when restricted to $H$-free graphs for particular choices of $H$. Therefore, these graph classes are well-studied throughout a range of NP-hard problems. In the context of coloring, e.g., 3-colorability is polynomially solvable for $P_6$-free graphs, while 4-colorability remains NP-hard for $P_{12}$-free graphs, and 5-colorability remains NP-hard for $P_8$-free graphs. We refer the reader to the survey paper [19] for more details. Note that also well-studied graph classes like chordal graphs (or, more generally, perfect graphs) and line graphs (or, more generally, claw-free graphs) can be characterized by forbidden subgraph conditions.

## 2  Results of this paper

One of the main open problems concerning on-line competitive coloring algorithms [7] is to decide whether for every $k$ there exists an on-line competitive coloring algorithm for the family of graphs with on-line chromatic number $k$. Perhaps surprisingly, this is even open for bipartite graphs for $k = 4$, whereas it has been solved for general graphs for $k \leq 3$: in both [8] and [17] it is proven that for the family of graphs with on-line chromatic number 3 at most 4 colors are needed. The open problem on bipartite graphs seems to be very hard and emphasizes how much on-line coloring differs from off-line coloring.

Our results are motivated by a number of open problems, but most strongly by the above open problem for bipartite graphs. We solve the problem for several subclasses of bipartite graphs which are defined by forbidding a certain fixed bipartite graph $H$ as an induced subgraph. For a relatively small graph $H$ this is an easy exercise, but for larger graphs this gets difficult, in correspondence with the fact that the class of $H$-free graphs contains the class of $H'$-free graphs if $H'$ is a subgraph of $H$. By combining known results and dealing with a few cases ourselves, we show that for every bipartite graph $H$ with at most 5 vertices there exists an on-line competitive coloring algorithm for the class of $H$-free bipartite graphs. Since for $P_4$-free and $P_5$-free graphs there even exists a competitive algorithm [9, 11], and since $P_6$-free bipartite graphs do not admit a competitive algorithm [9], our natural starting point from there is the latter class. In [3] we proved that there exists an on-line competitive algorithm for the class of $P_6$-free bipartite graphs; its bounding function is *linear* in the on-line chromatic number, namely roughly twice the on-line chromatic number. In fact, this gives a 2-approximation algorithm for on-line coloring $P_6$-free bipartite graphs. In this paper we prove a similar result for the larger class of $P_7$-free bipartite

3

graphs with a bounding function that is roughly eight times the on-line chromatic number. Note that the on-line chromatic number for both these graph classes can be arbitrarily high, so these classes are definitely no subclasses of the class of bipartite graphs with on-line chromatic number 4. In this sense, our results have a broader appeal than just solving the aforementioned problem with $k = 4$ for the restricted classes of $P_6$-free and $P_7$-free bipartite graphs. It might be possible that our algorithm or variations on it can be used to prove similar results for larger subclasses of bipartite graphs, although we have not been able to do so yet. We also note that our algorithm uses at most four colors (so is competitive) for the class of $P_5$-free bipartite graphs.

The rest of the paper is organized as follows. Section 3 contains the basic notation and definitions. In Section 4 we start our exposition by proving the result on $H$-free bipartite graphs with $|V(H)| \leq 5$. Next we introduce the key algorithm of this paper called *BicolorMax*. We prove in Section 6 that it is on-line competitive for $P_7$-free bipartite graphs. In Section 7 we deduce new upper bounds for the on-line chromatic number of bipartite graphs. We give some final remarks in Section 8 and conclusions in Section 9.

# 3    Preliminaries

Throughout we consider simple graphs $G = (V(G), E(G))$, where $V(G)$ is a set of vertices and $E(G)$ is a set of unordered pairs of vertices, called edges. For graph terminology not defined below we refer to [1].

A graph is called *bipartite* if its vertex set can be partitioned into two disjoint sets so that all edges are incident with vertices from both sets. If $H$ is a *subgraph* of $G$, i.e., if $H$ is a graph and $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$, we write $H \subseteq G$. If $S \subseteq V(G)$, then $G[S]$ denotes the subgraph of $G$ with vertex set $S$ and edge set $\{\{x, y\} \mid x \in S, y \in S\}$. A graph is an *induced subgraph* of $G$ if it is isomorphic to $G[S]$ for some nonempty $S \subseteq V(G)$. A graph is *H-free* if it does not contain the graph $H$ as an induced subgraph. We call two vertex-disjoint graphs *remote* if there are no edges joining them. A maximal connected subgraph of a graph $G$ is called a *component* of $G$. For any two vertices $x, y$ of a connected graph $G$ we denote by $P_{xy}$ a path between $x$ and $y$ in $G$. We define the *distance* $d(x, y, G)$ between $x$ and $y$ in $G$ as the number of edges of a shortest path between $x$ and $y$. We use $K_n$, $C_n$ and $P_n$ to denote, respectively, the complete graph, the cycle and the path on $n$ vertices, and we use $K_{m,n}$ to denote the complete bipartite graph with $m$ vertices in one bipartition class and $n$ vertices in the other. A *coloring* of a graph $G$ is a function $c : V(G) \rightarrow \{1, 2, \ldots\}$ such that $c(v) \neq c(w)$ whenever $\{v, w\} \in E(G)$. The smallest number of colors in a coloring of $G$ is the *chromatic number* of $G$ and denoted by $\chi(G)$. Clearly, a graph $G$ is bipartite if and only if $\chi(G) \leq 2$.

We assume that the reader is familiar with the basic concept of an on-line coloring algorithm. For details we refer to [14]. Intuitively, an on-line coloring algorithm properly colors the vertices of a graph one by one, consistently using a fixed strategy, depending only on the subgraph induced by the revealed vertices and the colors that have been assigned to them by the algorithm, according to an externally determined ordering of the presented vertices.

A popular informative way of looking at on-line coloring is as a two-person game. In the two-person game, the *Drawer* reveals the vertices of $G$ one by one together with their adjacencies to vertices already revealed. The *Painter* irrevocably assigns an admissible color to the currently revealed vertex, based on a strategy involving the graph induced by the already revealed vertices and their colors, and the adjacencies of the currently revealed vertex. The aim of Drawer is to

reveal the vertices in an order that forces Painter to use many colors. The aim of Painter is to invent a strategy that uses as few colors as possible. The order in which Drawer reveals the vertices of $G$ is an ordering of the vertices of $V(G)$. Any strategy Painter chooses can be extended to some on-line coloring algorithm in the above sense that assigns Painter's coloring to $G$. The reverse is also true: a coloring prescribed by an on-line coloring algorithm can be simulated by a two-person game with a Drawer and Painter as described above.

We denote the (finite) set of all on-line coloring algorithms for a graph $G$ by $AOL(G)$. Let $\Pi(G)$ denote the set of all permutations of the vertices of $G$. If $A \in AOL(G)$ and $\pi \in \Pi(G)$, we denote by $\chi_A(G, \pi)$ the number of colors used by $A$ when the vertices of $G$ are presented according to $\pi$. The largest number of colors used by the on-line algorithm $A$ for $G$ is called the $A$-chromatic number of $G$ and denoted by $\chi_A(G)$. Hence

$$\chi_A(G) = \max_{\pi \in \Pi(G)} \chi_A(G, \pi).$$

The smallest number of colors used by an on-line algorithm for $G$ is the *on-line chromatic number* of $G$, and denoted by $\chi_{OL}(G)$ [9]. Hence

$$\chi_{OL}(G) = \min_{A \in AOL(G)} \chi_A(G).$$

Let $\mathcal{G}$ denote a (possibly infinite) family of graphs. If $A \in AOL(G)$ for every $G \in \mathcal{G}$, we say that $A$ is an on-line coloring algorithm for $\mathcal{G}$ and write $A \in AOL(\mathcal{G})$. An algorithm $A \in AOL(\mathcal{G})$ is said to be *competitive* for $\mathcal{G}$ if there exists a function $f$ such that $\chi_A(G) \leq f(\chi(G))$ for every $G \in \mathcal{G}$; it is *on-line competitive* if $\chi_A(G) \leq f(\chi_{OL}(G))$ for every $G \in \mathcal{G}$.

# 4    Small forbidden subgraphs

As stated before, it has been shown that there does not exist a competitive on-line coloring algorithm for $P_6$-free bipartite graphs, but there exists a competitive on-line coloring algorithm for $P_5$-free bipartite graphs. In fact, combining results from [7, 11, 15, 16], and analyzing a few cases ourselves, we can show there exists an on-line coloring algorithm that is on-line competitive for the class of $H$-free bipartite graphs for any fixed bipartite graph $H$ on at most five vertices. We are not able to show such a result for any fixed graph $H$ that is not bipartite. Since a bipartite graph does not contain a non-bipartite subgraph, this would come down to proving that there exists an on-line coloring algorithm that is on-line competitive for the class of bipartite graphs. This however is still a major open problem.

Before stating and proving our proposition below, let us first note that we are not aiming at obtaining nice performance ratios in our proof of this proposition. As we are primarily interested in the existence result here we took the freedom to use rather unsophisticated methods for reaching our goal.

**Proposition 1** *Let $H$ be a bipartite graph on at most five vertices. Then there exists an on-line coloring algorithm that is on-line competitive for the class of $H$-free bipartite graphs.*

*Proof.* We may restrict ourselves to bipartite graphs on exactly five vertices, noting that an $F$-free bipartite graph with $F$ bipartite on at most four vertices is also $H$-free for some bipartite graph $H$ on five vertices. We use $H + H'$ to denote the disjoint union of two graphs $H$ and $H'$, and $pH$ to denote the disjoint union of $p \geq 2$ copies of $H$. Before we make a case distinction we first make the following easy observation:

5

(1) Let $F$ be a graph and $A$ an on-line coloring algorithm that is on-line competitive for the class of $F$-free bipartite graphs. Then there exists an on-line coloring algorithm $A'$ that is on-line competitive for the class of $F + K_1$-free bipartite graphs.

This claim can be seen as follows. Initially we use algorithm $A$ to color the vertices of an $F + K_1$-free bipartite graph $G$. If $G$ does not contain an induced $F$, then we do not need any extra colors. Suppose $G$ contains an induced subgraph $G'$ that is isomorphic to $F$. We assume that $G'$ is the first occurrence of $F$ if $G$ is revealed to $A$. Let $v$ be the last vertex of $G'$ presented to $A$. We color $v$ with a new color $c^*$. Since $G$ is $F + K_1$-free, any vertex $w$ that is presented after $v$ must have a neighbor in $G'$. Let $Q_1, \ldots, Q_p$ be the components of $G'$. We note that these components are bipartite, since $G$ is bipartite. For each component $Q_i$ $(i = 1, \ldots, p)$ we define a set of two new colors $\{c_i, d_i\}$, each of which corresponds to one of the classes in the bipartition of $Q_i$. Now suppose $Q_h$ is the component that has the smallest index $h$ of all the components of $G'$, which $w$ is adjacent to. Then we color $w$ with $c_h$ or $d_h$ depending on which bipartite class of $Q_h$ vertex $w$ belongs to. This way we can finish the coloring of $G$ with at most $2p + 1$ extra colors.

We now distinguish a number of cases depending on the value of $|E(H)| = m$.

*Case I*: $m = 0$. Then $H = 5K_1$ and consequently $|V(G)| \leq 8$. Then, since $G$ is bipartite, $\Delta(G) \leq 4$. Hence we obtain $\chi_{FF} \leq \Delta(G) + 1 \leq 5$.

*Case II*: $m = 1$. Then $H = K_2 + 3K_1$. It is trivial to see that $FF$ is on-line competitive for the class of $K_2$-free graphs, i.e., graphs with only isolated vertices. After applying (1) three times we get the desired result.

*Case III*: $m = 2$. Then $H = P_3 + 2K_1$ or $2K_2 + K_1$. For the first subcase we can proceed similarly as in Case II. For the second subcase we use the following result from [11]:

(2) If $G$ is a $P_5$-free graph without triangles, then $\chi_{FF}(G) \leq 3$.

Noting that $2K_2$-free bipartite graphs are both $P_5$-free and triangle-free, and combining (1) and (2), yields the result.

*Case IV*: $m = 3$. Then $H = P_4 + K_1$, $K_{1,3} + K_1$, or $P_3 + K_2$. Noting that $P_4$-free bipartite graphs are both $P_5$-free and triangle-free, and combining (1) and (2), yields the desired result for the first subcase. For the second subcase we first observe that $\chi_{FF}(G) \leq \Delta(G) + 1 \leq 3$ for any $K_{1,3}$-free bipartite graph $G$ and then we apply (1) to get the result. Since a $P_3 + K_2$-free bipartite graph is a $P_7$-free bipartite graph, we can of course immediately apply Theorem 10 (which will be presented later) for the third subcase. It is also not difficult to give a direct proof that our algorithm *BicolorMax* (which will be presented in the next section) is on-line competitive for this class of graphs.

*Case V*: $m = 4$. Then $H = K_{1,4}$, $C_4 + K_1$, $P_5$, or the unique graph with degree sequence $3,2,1,1,1$ which we denote by $K_{1,3}^+$. For the first subcase we easily get that $\chi_{FF}(G) \leq \Delta(G) + 1 \leq 4$. The *girth* of a graph $G$ is the number of edges of a smallest cycle in $G$. For the second subcase we combine (1) with the following result from [7]:

(3) If $G$ has girth at least five, then $\chi_{FF}(G) \leq \binom{2\chi_{OL}(G)}{2}$.

For the third subcase we use (2). The *radius* of a graph $G$ is defined as the minimum of $\max_v d(u, v, G)$ over all vertices $u$ in $G$. Since $K_{1,3}^+$ has radius 2, we can use the following result from [16]:

(4) For every tree $T$ with radius 2, there is an on-line coloring algorithm $A$ that is on-line competitive for the class of $T$-free graphs.

*Case VI*: $m = 5$. Then $H = K_{2,3} - e$ for an edge $e$ of $K_{2,3}$. We need a separate proof for this case. We first prove the following claim:

*Claim:* Let $G$ be bipartite and $H$-free and let $C$ be a component of $G$ such that $C_4$ is an induced subgraph of $C$. Then $C = K_{s,t}$ for some integers $s, t \geq 2$.

We prove this claim as follows. If $C = C_4 = K_{2,2}$, then the claim trivially holds. If not, let $C_4 = uvwxu$, and let $N(p)$ denote the neighbors of vertex $p$ in $C$. If $N(u) \not\subseteq N(w)$, then $G$ contains $H$ as an induced subgraph. So, by symmetry, $N(u) = N(w)$, and similarly $N(v) = N(x)$. Let $y \in N(u) \cap N(w)$. Then $uvwyu$ is an induced $C_4$, so as before $N(y) = N(v) = N(x)$. Hence all neighbors of $u$ and $w$ are adjacent to all neighbors of $v$ and $x$, and vice versa. By repeating the arguments for all induced $C_4$s, we obtain that $C = K_{s,t}$ for some $s, t \geq 2$.

A component in a bipartite graph that does not contain an induced $C_4$ has girth at least six. Since $\chi_{FF}(K_{s,t}) = 2$, the above claim together with (3) then implies that $\chi_{FF}(G) \leq \max\{\binom{2\chi_{OL}(G)}{2}, 2\}$.

*Case VII*: $m = 6$. Then $H = K_{2,3}$. Kierstead and Penrice [15] showed that $FF$ is on-line competitive for the class of $H$-free graphs. $\blacksquare$

We conclude that the first open question with respect to the (non)existence of on-line competitive coloring algorithms for $H$-free bipartite graphs concerns bipartite graphs $H$ on 6 vertices, in particular $H = P_6$, which has radius 3. In the next section we present an on-line algorithm for coloring general bipartite graphs. We will show that it is on-line competitive for ($P_6$-free and) $P_7$-free bipartite graphs.

## 5    The algorithm BicolorMax

Let $G$ be a bipartite graph on $n$ vertices denoted by $1, 2, \ldots, n$. Let $A = \{a_1, a_2, \ldots, a_p\}$ and $B = \{b_1, b_2, \ldots, b_p\}$ be two disjoint ordered sets of colors. For a fixed positive integer $k \leq p$, let $A(k) = \{a_1, a_2, \ldots, a_k\}$ and $B(k) = \{b_1, b_2, \ldots, b_k\}$.

We first give the general idea of our on-line algorithm called *BicolorMax*. Suppose that $G$ is presented to the algorithm. At some stage a new uncolored vertex $v$ of $G$ is revealed, together with its adjacencies to the set $S$ of already colored vertices of $G$. If $v$ is not adjacent to any previously revealed vertex of $G$, then $v$ receives color $a_1$. Otherwise, the choice of the color for $v$ is based on the present colors in the bipartition classes of the component containing $v$ of the subgraph of $G$ induced by $v$ and the vertices of $S$ with colors in $A(k) \cup B(k)$ for some suitable $k \geq 1$. To make this choice explicit we first need to introduce some additional terminology.

If $F \subseteq V(G)$, then the *hue* of $F$, denoted by $H(F)$, is the set of all colors used on vertices in $F$. Let $\pi \in \Pi(G)$ for a bipartite graph $G$, and assume that $v = \pi(j)$. Let $G_j^k$ denote the subgraph of $G[\{\pi(1), \ldots, \pi(j)\}]$ induced by $v = \pi(j)$ and all the vertices in $\{\pi(1), \ldots, \pi(j-1)\}$ that have been assigned colors from $A(k) \cup B(k)$. We denote by $C_j^k$ the component of $G_j^k$ containing $v$, and we write $C_j^k := (I_1, I_2)$ to indicate the bipartition of its vertex set. Note that $(I_1, I_2)$ is the unique bipartition of $C_j^k$ (up to renaming), because $C_j^k$ is connected. We say that color $a_k$ is *mixed* on $C_j^k = (I_1, I_2)$ if there exist at least two vertices $v \in I_1$ and $w \in I_2$ that have been colored with $a_k$. We then call $(v, w)$ a *k-mixed pair*.

7

We are now ready to explain the choice of the color of a new vertex. The key idea is to prevent the mixing of colors as much as possible. Recall that in the worst-case $FF$ uses $m$ colors for a complete bipartite graph $K_{m,m}$ minus a perfect matching, because the worst-case ordering forces $FF$ to assign the same color to each pair of the perfect matching, hence all $m$ colors are used on both bipartite classes. In order to reduce the total number of colors we somehow need to avoid this as much as possible by changing the coloring strategy with respect to mixing colors.

The algorithm $BicolorMax$ has been designed with this intuitive idea in mind. It is defined inductively. Vertex $\pi(1)$ is colored with $a_1$. Suppose that vertices $\pi(1), \dots, \pi(j-1)$ have already been colored. Let $v = \pi(j)$ be the next vertex presented to the algorithm. The algorithm first computes the highest value of $k$ such that $a_k$ is mixed on $C_j^k$. We denote this value by $m$. In order to stop the mixing of colors, the algorithm colors $v$ with $a_{m+1}$ or with $b_{m+1}$. By definition of $m$, the color $a_{m+1}$ is not mixed on $C_j^{m+1}$. The choice will fall on $a_{m+1}$ if $v$ is in the same bipartition class as a vertex with color $a_{m+1}$ or if $a_{m+1}$ has not been used before on $C_j^{m+1}$. Otherwise $v$ receives color $b_{m+1}$. This way the mixing of color $a_{m+1}$ within a component is prevented. We note that $a_{m+1}$ might get mixed later on: if at a certain stage two components that both contain a vertex with color $a_{m+1}$ are joined by a new vertex, then $a_{m+1}$ becomes mixed if at least two vertices with color $a_{m+1}$ belong to different bipartition classes.

Before we give a formal description of our algorithm we would like to make the following remark. The algorithm $FF$ uses at most three colors on any path $P_n$. However, $BicolorMax$ does not try to use the colors $a_k$ or $b_k$ for any $k < m$ if $a_m$ is mixed on $C_j^m$. Hence, it is easy to see that $BicolorMax$ is not on-line competitive for the family of paths. In Section 8 we return to this observation and show how we can combine $FF$ and $BicolorMax$ to obtain an on-line competitive algorithm for families of graphs for which $FF$ is on-line competitive as well.

---

$BicolorMax(G[\{\pi(1), \dots, \pi(j-1)\}], v)$

$m := \max(\{0\} \cup \{k : a_k \text{ is mixed on } C_j^k\})$.
**if** $a_{m+1} \notin H(V(C_j^{m+1}))$
    $C_j^{m+1} := (I_1, I_2)$ such that $v \in I_1$
**else**
    $C_j^{m+1} := (I_1, I_2)$ such that $a_{m+1} \in H(I_1)$.
**if** $v \in I_1$
    assign color $a_{m+1}$ to $v$
**else**
    assign color $b_{m+1}$ to $v$.

---

It is immediately clear that $BicolorMax$ is not able to color a non-bipartite graph $G$. For instance, take $G = K_3$. Then $C_3^1 = C_3^2 = K_3$ can not be partitioned into two independent sets $I_1$ and $I_2$. For bipartite graphs, $BicolorMax$ does assign a coloring for each vertex permutation.

**Observation 2** *$BicolorMax$ is an on-line coloring algorithm for bipartite graphs.*

*Proof.* Let $\pi \in \Pi(G)$ for a bipartite graph $G$. Let $v = \pi(j)$ be the $j^{th}$ vertex presented to $BicolorMax$. Let $m := \max(\{0\} \cup \{k : a_k \text{ is mixed on } C_j^k\})$.

Suppose $a_{m+1} \notin H(V(C_j^{m+1}))$. This means that $v$ is not adjacent to a previously presented vertex with color $a_{m+1}$. Hence, $BicolorMax$ may color $v$ with color $a_{m+1}$ in a proper vertex coloring.

Suppose $a_{m+1} \in H(V(C_j^{m+1}))$. By maximality of $m$, we find that $a_{m+1}$ is not mixed on $C_j^{m+1}$. Then, $a_{m+1}$ is only assigned to vertices in one bipartite class of $C_j^{m+1}$ and we can indeed define $C_j^{m+1} = (I_1, I_2)$ with $a_{m+1} \in H(I_1)$ and $a_{m+1} \notin H(I_2)$. If $v$ is in $I_1$, then $BicolorMax$ colors $v$ with $a_{m+1}$. Since $a_{m+1} \notin H(I_2)$, vertex $v$ is not adjacent to any previously presented vertex with color $a_{m+1}$. If $v$ is in $I_2$, then $BicolorMax$ colors $v$ with $b_{m+1}$. Suppose $v$ is adjacent to a vertex $u = \pi(i)$ for some $i \le j-1$ with color $b_{m+1}$. By definition of the algorithm, $C_i^{m+1} = (I_1', I_2')$ with $a_{m+1} \in H(I_1')$ and $u \in I_2'$. Since $C_i^{m+1} \subseteq C_j^{m+1}$, we find that $I_1' \subseteq I_1$ and $I_2' \subseteq I_2$. This leads to a contradiction because the vertices $u$ and $v$ can not be adjacent if they are both in $I_2$. ∎

To illustrate how the choice of the color of $v = \pi(j)$ depends on the components of $G_j^k$ we apply the algorithm to the following example.

**Example.** Let $G$ be a $K_{4,4}$ without a perfect matching, i.e., with $V(G) = \{1,2,3,4,5,6,7,8\}$, bipartition in $\{1,3,5,7\}$ and $\{2,4,6,8\}$, and only edges $\{1,2\}$, $\{3,4\}$, $\{5,6\}$, and $\{7,8\}$ omitted. If the vertices are revealed in the order of increasing numbers, the algorithm assigns colors $a_1, a_1, b_1, b_1, a_2, b_2, a_2, b_2$, respectively. The color $b_2$ is assigned to vertex 8 for the following three reasons. Firstly, $a_1$ is mixed on $C_8^1$, i.e., the subgraph of $G$ induced by $\{1,2,3,4,8\}$. Secondly, $a_2$ is not mixed on $C_8^2 = G$. Thirdly, $a_2$ has already been assigned to a vertex in $\{1,3,5,7\}$, i.e., the bipartition class of $C_8^2$ not containing the vertex 8.

Suppose that $G$ is extended and a new vertex 9 is revealed to the algorithm after presenting vertices $1, \ldots, 8$. Suppose 9 is only adjacent to 7. Then $a_1$ is not mixed on $C_9^1 = (\{9\}, \emptyset)$ and $a_2$ is not mixed on $C_9^2 = G$. Hence $BicolorMax$ colors 9 with $a_1$. It is left to the reader to check that 9 is assigned color $b_1$ if 9 is adjacent to 1 and 7, color $b_2$ if 9 is adjacent to 1, 3 and 7, and color $a_2$ if 9 is adjacent to 2, 4 and 6.

For a $K_{n,n}$ without a perfect matching with $n \ge 5$ the algorithm will continue assigning $a_2$ and $b_2$ if the vertices are presented in an order alternating between the two classes of the bipartition, as in the above example for $n = 4$. In contrast, recall that $FF$ uses $n$ colors in this case.

**Remark.** In [18], the authors leave it as an "easy exercise" to construct an on-line coloring algorithm $A$ such that $\chi_A(G) = O(\log_2 n)$ on the class of bipartite graphs (where $G$ is a bipartite graph on $n$ vertices). There are several choices for $A$ to solve this exercise. One of these choices is the algorithm $BicolorMax$ that we presented above and that will be analyzed in this paper. We are not aware of any publications that contain this algorithm $BicolorMax$ (except for [3] of course). Clearly, our main purpose in this paper is to show that $BicolorMax$ works particularly well in the sense of on-line competitiveness for $P_7$-free bipartite graphs. As can be concluded from the next section this is not an easy exercise.

One might be inclined to think that $BicolorMax$ coincides with other on-line algorithms that have been proposed to solve the "easy exercise" from [18], e.g., the algorithm from the survey paper of Kierstead (see [14], Theorem 2 on page 286) that is also known under the acronym $BFF$ (Bipartite First-Fit). The formulation of $BFF$ is very simple: when vertex $v_i$ is presented, there is a unique partition $(I_1, I_2)$ of the component to which $v_i$ belongs such that $v_i \in I_1$. The algorithm $BFF$ assigns to $v_i$ the smallest color (in some ordering of the colors) that has not already been assigned to a vertex of $I_2$. We do something more advanced, as can be seen from the above example. In this example, as we have argued, if the vertices 1 to 8 are presented in the order of increasing numbers, the algorithm $BicolorMax$ assigns colors $a_1, a_1, b_1, b_1, a_2, b_2, a_2, b_2$, respectively. It can be checked that $BFF$ also assigns these colors in the same order. Now suppose a new vertex 9 is presented which is only adjacent to vertex 7. Then $BicolorMax$ assigns color $a_1$ to 9, as explained

9

in the previous example, whereas *BFF* assigns color $b_2$ to 9.

Although at first sight *BicolorMax* looks very similar to *BFF*, the subtle difference in the choice of colors to avoid mixing colors enables us to prove the on-line competitiveness of our algorithm for $P_7$-free bipartite graphs. Our earlier attempts to use a simpler strategy, like in *BFF*, for such a competitiveness proof failed even for the more restricted class of $P_6$-free bipartite graphs. In fact, finding a suitable variant for our competitiveness proof turned out to be rather tricky.

# 6   $P_7$-free bipartite graphs

In this section we prove our main theorem showing that *BicolorMax* is a linear on-line competitive algorithm for the class of $P_7$-free bipartite graphs. The proof is modelled along the following lines. We will define a class of tree-like bipartite graphs, show that the on-line competitive factor of any on-line coloring algorithm for these graphs is high, and show that whenever *BicolorMax* uses many colors on a $P_7$-free bipartite input graph $G$, then $G$ contains a large member of the class of tree-like graphs as an induced subgraph. More specifically, denote by $\chi_{BM}(G)$ the maximum number of colors used by *BicolorMax* for coloring a graph $G$.

By a series of lemmas and propositions we prove that $\chi_{BM}(G) \leq 8\chi_{OL}(G) + 8$ for any $P_7$-free bipartite graph $G$. In the next two lemmas we first show that *BicolorMax* controls the mixing of color pairs. We note that these lemmas are valid for any bipartite graph.

**Lemma 3** *Let $G$ be a bipartite graph. Let BicolorMax color vertex $v = \pi(j)$ with $a_m$ or $b_m$, $m \geq 2$. Let $(x, y)$ be a $k$-mixed pair in $C_j^k$ with $k \leq m - 1$. Then any path between $x$ and $y$ in $C_j^k$ must pass through $v$.*

*Proof.* Let $x = \pi(r)$ and let $y = \pi(s)$ for some $r, s \leq j - 1$. We assume without loss of generality that $y$ has been presented to *BicolorMax* after $x$, i.e., $s > r$. Suppose $x$ belongs to $C_s^k$, implying that $a_k \in H(V(C_s^k))$. Since $y$ is colored with $a_k$, color $a_k$ is not mixed on $C_s^k$. Since $(x, y)$ is a $k$-mixed pair, $x$ (together with all the other vertices in $C_s^k$ that have color $a_k$) and $y$ are in different bipartite classes. So *BicolorMax* would have colored $y$ with color $b_k$. Hence $x$ does not belong to $C_s^k$.

Suppose there exists an index $i$ with $s < i < j$ such that $x$ and $y$ belong to $C_i^k$. This means that $a_k$ is mixed on $C_i^k$. Then *BicolorMax* would never use a color $a_h$ or $b_h$ with $h \leq k$ to color $\pi(i)$. We conclude that every path between $x$ and $y$ in $C_j^k$ must pass through $v$. ∎

**Lemma 4** *Let $G$ be a bipartite graph. Let BicolorMax color vertex $v = \pi(j)$ with $a_m$ or $b_m$, $m \geq 2$. For some $k \leq m - 1$ let $x$ and $y$ be two vertices in $C_j^k$ colored with $a_k$ and $b_k$ respectively such that $d(x, y, C_j^k)$ is even. Then any path between $x$ and $y$ in $C_j^k$ must pass through $v$.*

*Proof.* Let $x = \pi(r)$ and let $y = \pi(s)$ for some $r, s \leq j - 1$. Suppose $x$ belongs to $C_s^k$ implying that $y$ appeared after $x$. Since $y$ is colored with $b_k$, color $a_k$ is not mixed on $C_s^k$. Because $G$ is bipartite and the distance $d(x, y, C_j^k)$ is even, also the distance between $x$ (or any other vertex in $C_s^k$ with color $a_k$) and $y$ in $C_s^k$ is even. Then *BicolorMax* would have colored $y$ with color $a_k$. Hence $x$ does not belong to $C_s^k$.

Suppose $y$ belongs to $C_r^k$. Since $x$ is colored with $a_k$, color $a_k$ is not mixed on $C_r^k$. Then the distance between $x$ and any vertex in $C_r^k$ with color $a_k$ is even. Since $y$ has received color $b_k$, by definition of *BicolorMax*, there exists at least one vertex $z$ in $C_r^k$ with color $a_k$, which is in a

10

different bipartite class than $y$. Since the distance between $x$ and $y$ in $C_j^k$, and consequently in $C_r^k$, is even, this implies that the distance between $x$ and $z$ in $C_r^k$ is odd. This is a contradiction. Hence, $y$ does not belong to $C_r^k$.

In the remaining case there exists an index $i$ with $\max(\{r, s\}) < i < j$ such that $x$ and $y$ belong to $C_i^k$. Since $y$ is colored with $b_k$, by definition of *BicolorMax*, the component $C_s^k$, which is a subgraph of $C_i^k$, must contain a vertex $w$ with color $a_k$ and with odd distance to $y$. Since the distance between $x$ and $y$ is even, the distance between $w$ and $x$ is odd, and we find that $a_k$ is mixed on $C_i^k$. Then *BicolorMax* would never use a color $a_h$ or $b_h$ with $h \leq k$ to color $\pi(i)$. We conclude that every path between $x$ and $y$ in $C_j^k$ must pass through $v$. ■

In the following lemma we show that *BicolorMax* ensures that vertices of higher colors are very "close" to at least some vertices of lower colors.

**Lemma 5** *Let $G$ be a $P_7$-free bipartite graph. Let BicolorMax color vertex $v = \pi(j)$ with $a_m$ or $b_m$, $m \geq 3$. There exists an $(m-1)$-mixed pair $(z^*, z)$ in $C_j^{m-1}$ with $d(v, z^*, C_j^{m-1}) = 1$ and $d(v, z, C_j^{m-1}) = 2$.*

*Proof.* By definition of *BicolorMax*, color $a_{m-1}$ is mixed on $C_j^{m-1}$. So $C_j^{m-1}$ contains at least one $(m-1)$-mixed pair $(z, z')$. We assume without loss of generality that the distance between $z$ and $v$ is even and the distance between $z'$ and $v$ is odd. Since $(z, z')$ is an $(m-1)$-mixed pair, by Lemma 3, a shortest path $P_{zz'}$ from $z$ to $z'$ in $C_j^{m-1}$ must be formed by joining shortest paths $P_{zv}$ from $z$ to $v$ and $P_{vz'}$ from $v$ to $z'$. First we show that $d(v, z, C_j^{m-1}) = 2$.

Suppose $d(v, z, C_j^{m-1}) \geq 4$. Then $d(v, z', C_j^{m-1}) = 1$, i.e., $z'$ and $v$ are adjacent. Otherwise, since $z'$ has odd distance to $v$, the path $P_{zz'}$ contains an induced $P_7$. Let $z' = \pi(s)$ for some $s < j$. Since *BicolorMax* has used color $a_{m-1} \neq a_1$ (due to our assumption that $m \geq 3$) on vertex $z'$, the component $C_s^{m-2}$ contains an $(m-2)$-mixed pair. This means that $z'$ has a neighbor $w \neq v$ in $C_s^{m-2} \subseteq C_j^{m-1}$. Bipartiteness of $G$ and Lemma 3 imply that $w$ is not adjacent to any vertex in $P_{zv}$. This implies that $G$ contains an induced $P_7$, which is a contradiction. Hence $d(v, z, C_j^{m-1}) = 2$.

We now show that $d(v, z', C_j^{m-1}) \leq 3$. Suppose $d(v, z', C_j^{m-1}) \geq 5$. Then $P_{zz'}$ would contain an induced $P_7$. Hence $d(v, z', C_j^{m-1}) = 1$ or $d(v, z', C_j^{m-1}) = 3$. In the first case we are done. In the second case, i.e., if $d(v, z', C_j^{m-1}) = 3$, we will show that there exists a vertex $z^*$ on $P_{z'v}$ with color $a_{m-1}$ that is adjacent to $v$.

Let $y$ and $z^*$ be vertices of $G$ such that $P_{z'v} = z'yz^*v$. Note that the distance between $z^*$ and $z'$ is even. Then, due to Lemma 4, vertex $z^*$ has not been colored with $b_{m-1}$. We will show that $z^*$ has not been colored with any color from $A(m-2) \cup B(m-2)$ either. First note that any neighbor of $z'$ in $C_j^{m-1}$ is adjacent to $z^*$. Otherwise, we could extend the path $P_{zz'}$ on six vertices with one extra vertex, and $C_j^{m-1}$ would contain an induced $P_7$.

Let $r < j$ be chosen such that $z^* = \pi(r)$. Recall that $z' = \pi(s)$. We first consider the case $s > r$, i.e., vertex $z'$ has appeared after $z^*$. Since $z^*$ is adjacent to every neighbor of $z'$ in $C_s^{m-2} \subseteq C_j^{m-1}$ and $a_{m-2}$ is mixed on $C_s^{m-2}$, Lemma 3 prevents that $z^*$ is in $C_s^{m-2}$. Otherwise, $C_s^{m-2}$ would contain a path (using $z^*$) between two vertices $u_1$ and $u_2$ of an $(m-2)$-mixed pair $(u_1, u_2)$ in $C_s^{m-2}$ not going through $z'$. We already noted that $z^*$ has not received color $b_{m-1}$. Then, since $z^*$ is in $C_j^{m-1}$, vertex $z^*$ must have been colored with $a_{m-1}$.

Now assume $s < r$, i.e., vertex $z^*$ has appeared after $z'$. Every neighbor of $z'$ in $C_r^{m-2}$ is adjacent to $z^*$. Hence $a_{m-2}$ is not only mixed on $C_s^{m-2}$ but also on $C_r^{m-2}$. Since $b_{m-1}$ was not allowed and $z^*$ is in $C_j^{m-1}$, *BicolorMax* must have colored $z^*$ with $a_{m-1}$. ■
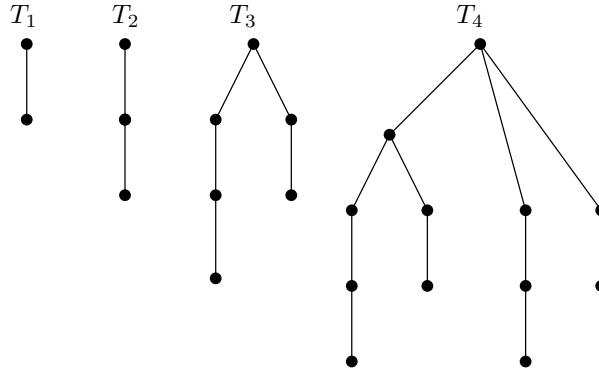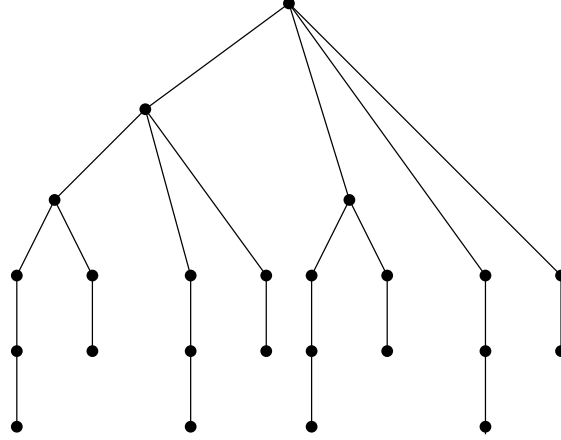
Figure 1: The trees $T_1, T_2, T_3, T_4$.



Figure 2: The tree $T_5$.

We inductively define a class of trees (see Figures 1 and 2). Each tree $T_k$ of the class has a root vertex $r(T_k)$, and:

- $T_1$ is a tree consisting of an edge, one of whose end vertices is the root vertex $r(T_1)$.

- $T_2$ is a path on three vertices, one of whose end vertices is the root vertex $r(T_2)$.

- $T_k$, $k \geq 3$ consists of a root vertex $r(T_k)$ that is adjacent to the root vertices of mutually disjoint copies of $T_1, T_2, \ldots, T_{k-1}$ (one copy of each of these trees). These copies are then called the *child trees* of $T_k$.

We denote a copy of a tree $T_k$ with root vertex $v = r(T_k)$ by $T_k(v)$. The child trees of $T_k(v)$ are denoted by $T_1^v, T_2^v, \ldots, T_{k-1}^v$. The following lemma turns out to be useful.

**Lemma 6** *Let $G$ be a $P_7$-free bipartite graph. If BicolorMax uses color $a_k$ or $b_k$ on vertex $v = \pi(j)$ with $k \geq 2$, then $C_j^{k-1}$ contains the tree $T_{k-1}(v)$ as a (not necessarily induced) subgraph in such a way that:*

*(i) If there exists an edge in $G$ between any two vertices $x, y$ in $T_{k-1}(v)$ with $d(v, x, T_{k-1}(v)) \leq d(v, y, T_{k-1}(v))$, then $x$ lies on the path from $y$ to $v$ in $T_{k-1}(v)$.*

*(ii) The root of child tree $T_i^v$ is colored with $a_{i+1}$ or $b_{i+1}$ for all $1 \le i \le k - 2$.*

*Proof.* By induction on $k$. Let $k = 2$, i.e., *BicolorMax* uses color $a_2$ or $b_2$ on vertex $v$. Then $C_j^1$ contains a 1-mixed pair. This implies that $v$ has a neighbor in $C_j^1$, and the conditions of the lemma are trivially satisfied.

Let $k = 3$, i.e., *BicolorMax* uses color $a_3$ or $b_3$ on vertex $v$. By Lemma 5, vertex $v$ has a neighbor $z^*$ in $C_j^2$ with color $a_2$. Let $z^* = \pi(q)$ for some $q < j$. Then $C_q^1$ contains a 1-mixed pair. This implies that $z^*$ has a neighbor not equal to $v$ in $C_j^2$. We conclude that the conditions of the lemma are satisfied.

Let $k \ge 4$. Since *BicolorMax* uses color $a_k$ or $b_k$ on vertex $v$, there exists a $(k-1)$-mixed pair $(x, y)$ in $C_j^{k-1}$. By Lemma 5, we may without loss of generality assume that $d(v, x, C_j^{k-1}) = 2$ and $d(v, y, C_j^{k-1}) = 1$. Assume $x = \pi(h)$ for some $h < j$ and $y = \pi(i)$ for some $i < j$. By the induction hypothesis, $C_h^{k-2}$ contains the tree $T_{k-2}(x)$, and $C_i^{k-2}$ contains the tree $T_{k-2}(y)$. Since $(x, y)$ is a $(k-1)$-mixed pair in $C_j^{k-1}$, every path from $x$ to $y$ in $C_j^{k-1}$ must go through $v$ due to Lemma 3. This implies that every path in $C_j^{k-1}$ from a vertex in $C_h^{k-2} \subseteq C_j^{k-1}$ to a vertex in $C_i^{k-2} \subseteq C_j^{k-1}$ must go through $v$. Then we have also found that every path in $C_j^{k-1}$ from a vertex in $T_{k-2}(x) \subseteq C_h^{k-2}$ to a vertex in $T_{k-2}(y) \subseteq C_i^{k-2}$ must go through $v$. We distinguish two cases: either $C_h^{k-2}$ contains a common neighbor of $v$ and $x$, or $C_h^{k-2}$ does not contain any common neighbors of $v$ and $x$.

**Case 1.** Component $C_h^{k-2}$ contains a common neighbor $w$ of $x$ and $v$. Again we need to distinguish two cases: either $w$ is in $T_{k-2}(x)$, or $w$ is not in $T_{k-2}(x)$.

**Case 1a.** Vertex $w$ is in $T_{k-2}(x)$. Then $w$ is in a child tree $T_p^x$ of $T_{k-2}(x)$ for some $1 \le p \le k - 3$. If $k = 4$, then $T_{k-2}(x) = T_2(x)$, and $w$ must be the root of $T_1^x$. By the induction hypothesis $w$ is colored with $a_2$ or $b_2$. Recall that every path in $C_j^3$ from a vertex in $T_2(x)$ to a vertex in $T_2(y)$ goes through $v$, and that $y$ is colored with $a_{k-1} = a_3$. This implies that $v$ is the root of a copy of $T_3$ satisfying (i) and (ii).

Suppose $k \ge 5$. Then $T_{k-2}(x)$ has a child tree $T_q^x$ with root $u$ for some $q \ne p$. Note that for all $k \ge 2$ any child tree of a tree $T_k$ consists of at least two vertices. Let $u'$ be a neighbor of $u$ in $T_q^x$. Let $y'$ be a neighbor of $y$ in $T_{k-2}(y)$. Note that $u', u, x, w, v, y, y'$ are seven different vertices of $G$. This implies that $P_{u'y'} = u'uxwvyy'$ is a path on seven vertices in $C_j^{k-1}$. Recall that every path from a vertex of $T_{k-2}(x)$ to a vertex of $T_{k-2}(y)$ goes through $v$. Then there are no edges between $\{u', u, x, w\}$ and $\{y, y'\}$. By the induction hypothesis $T_p^x$ and $T_q^x$ are remote. This implies that $w$ is neither adjacent to $u$ nor to $u'$. Then there must be an edge between $u$ and $v$, otherwise $P_{u'y'}$ is an induced $P_7$ in $G$. Hence, we have found that $v$ is adjacent to the root of all child trees of $T_{k-2}(x)$ that are not equal to $T_p^x$. However, also the root of $T_p^x$ must be joined to $v$ by an edge. This can be shown by using exactly the same arguments (in which vertex $u$ takes over the role of vertex $w$).

From the above we conclude that $v$ is adjacent to the roots of all child trees of $T_{k-2}(x)$. These trees together with tree $T_{k-2}(y)$ form the child trees of $T_{k-1}(v)$. Due to the fact that any path from a vertex in $T_{k-2}(x)$ to a vertex in $T_{k-2}(y)$ goes through $v$ and our induction hypothesis, the child trees of $T_{k-1}(v)$ satisfy (i). Recall that the root vertex $y$ of $T_{k-2}^v = T_{k-2}(y)$ is colored with $a_{k-1}$. The root vertices of the other child trees of $T_{k-1}(v)$ are colored with the desired colors due to the induction hypothesis. Hence, also condition (ii) of the lemma is satisfied.

**Case 1b.** Vertex $w$ is not in $T_{k-2}(x)$. Since $k \ge 4$, there exists a vertex $s$ with color $a_{k-2}$ in $C_h^{k-2}$ with $d(x, s, C_h^{k-2}) = 2$ due to Lemma 3. Since $d(x, w, C_h^{k-2}) = 1$, vertex $s$ is not equal to vertex $w$.

Again, let $y'$ be a neighbor of $y$ in $T_{k-2}(y)$. We first show that we may without loss of generality assume that $w$ is adjacent to $s$.

Suppose $w$ is not adjacent to $s$. Since $d(x, s, C_h^{k-2}) = 2$, vertex $s$ and $x$ have a common neighbor $t$ in $C_h^{k-2}$. Note that $s, t, x, w, v, y, y'$ are seven different vertices of $G$. This implies that $P_{sy'} = stxwvyy'$ is a path on seven vertices in $C_j^{k-1}$. Recall that every path from a vertex in $C_h^{k-2}$ to a vertex in $T_{k-2}(y)$ goes through $v$. Then there are no edges between $\{s, t, x, w\}$ and $\{y, y'\}$. This together with $\{w, s\} \notin E(G)$ implies that $v$ must be adjacent to $t$, otherwise $P_{sy'}$ is an induced $P_7$ in $G$. Then, we can pick vertex $t$ instead of vertex $w$. So from now on we assume that $w$ is a common neighbor of $s$ and $x$ in $C_h^{k-2}$.

Let $r$ be the root of child tree $T_{k-3}^x$ of $T_{k-2}(x)$. Let $r'$ be a neighbor of $r$ in $T_{k-3}^x$. Note that $r', r, x, w, v, y, y'$ are seven different vertices of $G$. This implies that $P_{r'y'} = r'rxwvyy'$ is a path on seven vertices in $C_j^{k-1}$. By the induction hypothesis $r$ is colored with $a_{k-2}$ or $b_{k-2}$.

Suppose $r$ is colored with $a_{k-2}$. Since $d(r, s, C_h^{k-2})$ is odd and $s$ is colored with $a_{k-2}$, the pair $(r, s)$ is a $(k-2)$-mixed pair in $C_h^{k-2}$. Lemma 3 implies that every path from $s$ in $C_h^{k-2}$, and hence every path from $w$ in $C_h^{k-2}$, to a vertex in $T_{k-3}^x \subseteq C_h^{k-2}$ goes through $x$. Then there are no edges between $\{r, r'\}$ and $w$. Furthermore, recall that every path in $C_j^{k-1}$ from a vertex in $C_h^{k-2}$ to a vertex in $T_{k-2}(y) \subseteq C_i^{k-2}$ goes through $v$. Then there are no edges between $\{r', r, x, w\}$ and $\{y, y'\}$ either. Since $P_{r'y'}$ may not be an induced $P_7$, these observations imply that there must be an edge between $v$ and $r$. Hence, using vertex $r$ instead of $w$ brings us back to Case 1a.

Suppose $r$ is colored with $b_{k-2}$. By Lemma 5, $C_h^{k-2}$ contains a vertex $\hat{s}$ that has received color $a_{k-2}$ and that is adjacent to $x$. Let $\hat{s} = \pi(\ell)$ for some $\ell < j$. By our induction hypothesis, vertex $\hat{s}$ is the root of a tree $T_{k-3}(\hat{s})$ in $C_\ell^{k-3}$.

Suppose every path in $C_h^{k-2}$ from $\hat{s}$ to a vertex in any child tree $T_p^x$ for $1 \leq p \leq k-4$ goes through $x$. Then in $T_{k-2}(x)$ we can replace the child tree $T_{k-3}(r)$ by the child tree $T_{k-3}(\hat{s})$. Then we can repeat the argument above (replace $r$ by $\hat{s}$, which is colored by $a_{k-2}$) in order to find that $v$ is adjacent to $\hat{s}$, and we return to Case 1a.

Suppose $C_h^{k-2}$ contains a path from $\hat{s}$ to a child tree $T_m^x$ for some $1 \leq m \leq k-4$ that does not go through $x$. Let $z$ be the root of $T_m^x$ and let $z'$ be a neighbor of $z$ in $T_m^x$. Note that $z', z, x, w, v, y, y'$ are seven different vertices of $G$. This implies that $P_{z'y'} = z'zxwvyy'$ is a path on seven vertices in $C_j^{k-1}$. Recall that $w$ is a neighbor of $s$. We then find that $z'$ and $w$ are not adjacent. Otherwise $C_h^{k-2}$ would contain a path between $s$ and $\hat{s}$, which are the vertices of a $(k-2)$-mixed pair, not going through $x$ (a contradiction due to Lemma 3). Then as before, in order not to have an induced $P_7$, vertex $v$ must be adjacent to $z$. We return to Case 1a.

**Case 2** Component $C_h^{k-2}$ does not contain a common neighbor of $x$ and $v$. Since $x$ has distance two from $v$ in $C_j^{k-1}$, there exists a common neighbor $v'$ of $v$ and $x$ in $C_j^{k-1}$. We first prove that $v'$ has received color $b_{k-1}$.

Since $k \geq 4$, Lemma 5 implies that $C_h^{k-2}$ contains a vertex $s$ with color $a_{k-2}$ at distance $d(x, s, C_h^{k-2}) = 2$ from $x$, and $C_h^{k-2}$ contains a vertex $t$ with color $a_{k-2}$ that is a neighbor of $x$. Since $d(x, s, C_h^{k-2}) = 2$, vertex $s$ and $x$ have a common neighbor $s'$ in $C_h^{k-2}$. Since $k \geq 4$, vertex $t$ has a neighbor $t'$ in $C_h^{k-2}$ that is not equal to $x$. Let $y'$ be a neighbor of $y$ in $T_{k-2}(y)$. Note that $s, s', t, t', x, v', v, y, y'$ are nine different vertices in $G$. This implies that both $P_{sy'} = ss'xv'vyy'$ and $P_{t'y'} = t'txv'vyy'$ are paths on seven vertices in $C_j^{k-1}$. Any path in $C_j^{k-1}$ from a vertex in $T_{k-2}(y)$ to a vertex in $C_h^{k-2}$ or to $v'$ goes through $v$. Otherwise there exists a path in $C_j^{k-1}$ from $y$ to $x$ that does not use $v$, which is not possible due to Lemma 3. Hence, there are no edges between

$\{s, s', x, v'\}$ and $\{y, y'\}$, and there are no edges between $\{t, t', x, v'\}$ and $\{y, y'\}$ either. Since we assumed that $C_h^{k-2}$ does not contain any common neighbor of $v$ and $x$, there are no edges between $v$ and $\{s', t\}$. Then $v'$ must be adjacent to both $s$ and $t'$, otherwise $G$ contains an induced $P_7$. Since $v'$ is in $C_j^{k-1}$ and adjacent to vertex $x$ with color $a_{k-1}$, the color of $v'$ is in $A(k-2) \cup B(k-1)$.

Suppose $v'$ has not received color $b_{k-1}$ but some color from $A(k-2) \cup B(k-2)$. Then $v'$ must have appeared after $x$, due to our assumption that $v'$ is not in $C_h^{k-2}$. However, in that case, $v'$ has also appeared after $s$ and $t$. Then $(s, t)$ is a $(k-2)$-mixed pair of $C_{\pi^{-1}(v')}^{k-2}$ implying that $BicolorMax$ would never color $v'$ with a color from $A(k-2) \cup B(k-2)$. Hence, $v'$ must have received color $b_{k-1}$.

Recall that $y = \pi(i)$ is assigned color $a_{k-1}$. Then, by Lemma 5, component $C_i^{k-2}$ contains a $(k-2)$-mixed pair $(z^*, z)$ such that $d(y, z^*, C_i^{k-2}) = 1$ and $d(y, z, C_i^{k-2}) = 2$. We first show that $v$ is adjacent to $z$ and every neighbor of $z^*$ in $C_i^{k-2}$.

Since $d(y, z, C_i^{k-2}) = 2$, component $C_i^{k-2}$ contains a common neighbor $z'$ of $y$ and $z$. Let $x'$ be a neighbor of $x$ in $C_h^{k-2}$. Since $v'$ with color $b_{k-1}$ and neighbor $v$ with color $a_{k-1}$ are neither in $C_h^{k-2}$ nor in $C_i^{k-2}$, we find that $z, z', y, v, v', x, x'$ are seven different vertices. This implies that $P_{zx'} = zz'yvv'xx'$ is a path on seven vertices in $C_j^{k-1}$. By Lemma 3, any path from $y$ to $x$ in $C_j^{k-1}$ must go through $v$. This implies that there are no edges between vertices from $\{z, z', y\}$ and $\{v', x, x'\}$. Since we assume that $v$ and $x$ do not have a common neighbor in $C_h^{k-2}$, vertices $x'$ and $v$ are not adjacent. Then $v$ must be adjacent to $z$. By the same arguments we find that $v$ is adjacent to every neighbor of $z^*$ in $C_i^{k-2}$.

Let $z^* = \pi(\ell^*)$ for some $\ell^* < j$. By our induction hypothesis, $C_{\ell^*}^{k-3}$ contains the tree $T_{k-3}(z^*)$ satisfying conditions (i) and (ii) of the lemma. Since $v$ is adjacent to every neighbor of $z^*$ in $C_i^{k-2} \supseteq C_{\ell^*}^{k-3}$, vertex $v$ is adjacent to the roots of all child trees $T_{k-4}^{z^*}, T_{k-5}^{z^*}, \ldots, T_1^{z^*}$.

We are also going to use our induction hypothesis with respect to vertex $z$, which has been assigned color $a_{k-2}$. Let $z = \pi(\ell)$ for some $\ell < j$. Then $C_\ell^{k-3}$ contains the tree $T_{k-3}(z)$ satisfying conditions (i) and (ii) of the lemma. Recall that $z$ is adjacent to $v$. So $v$ is adjacent to the root of a copy of a tree $T_{k-3}$. Due to Lemma 3, the trees $T_{k-3}(z)$ and $T_{k-3}(z^*)$ do not have a vertex in common.

Finally, we consider vertex $v'$ with color $b_{k-1}$. Let $v' = \pi(\ell')$ for some $\ell' < j$. Again by our induction hypothesis, $C_{\ell'}^{k-2}$ contains the tree $T_{k-2}(v')$ satisfying conditions (i) and (ii) of the lemma. Recall that $v$ is adjacent to $v'$. So $v$ is adjacent to a copy of $T_{k-2}$ that does not have any vertex in common with the trees $T_{k-3}(z)$ and $T_{k-3}(z^*)$ due to Lemma 3.

We conclude that $C_j^{k-1}$ contains the tree $T_{k-1}(v)$ as a subgraph such that condition (ii) has been satisfied and that condition (i) will be satisfied as well if we can show the following statement: there are no edges between $T_{k-3}^v = T_{k-3}(z)$ and a child tree $T_i^v = T_i^{z^*}$ for all $1 \le i \le k - 4$, and there are no edges between $T_{k-2}^v = T_{k-2}(v')$ and a child tree $T_i^v$ for all $1 \le i \le k - 3$. This claim can be seen as follows. If there is an edge between $T_{k-2}^v$ and a child tree $T_i^v$ for $1 \le i \le k - 3$, then $C_j^{k-1}$ contains a path from $x$ to $y$ that does not go through $v$. Since $(x, y)$ is a $(k-1)$-mixed pair in $C_j^{k-1}$, this is not possible due to Lemma 3. If there is an edge between $T_{k-3}^v$ and a child tree $T_i^v$ for $1 \le i \le k - 4$, then $C_i^{k-2}$ contains a path from $z$ to $z^*$ that does not go through $y$. Since $(z^*, z)$ is a $(k-2)$-mixed pair in $C_i^{k-2}$, this is not possible, again due to Lemma 3. ∎

We also inductively define a class of $P_6$-free bipartite graphs $F_i$ (see Figures 3 and 4). Its purpose and its relation to the class of trees $T_i$ will be made clear later on. Each graph $F_i$ of the class has a root vertex $r(F_i)$, and:

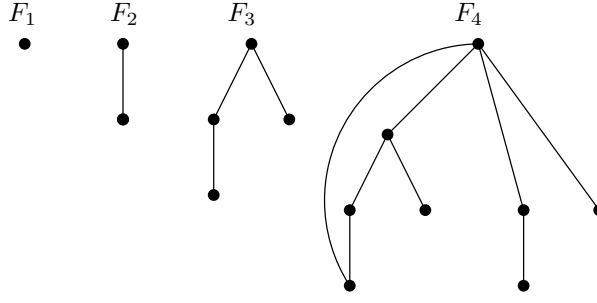- $F_1$ is a graph consisting of a single root vertex.

Figure 3: The graphs $F_1, F_2, F_3, F_4$.

- $F_2$ is a graph consisting of an edge, one of whose end vertices is the root vertex.

- $F_k$, $k \geq 3$ consists of a root vertex $r(F_k)$ that is adjacent to the root vertices of disjoint copies of $F_1, F_2, \ldots, F_{k-1}$ (one copy of each of these graphs). These copies are then called the *child graphs* of $F_k$. For all $1 \leq j \leq k - 1$, we join vertex $r(F_k)$ also to every vertex in $F_j$ that has distance 2 to $r(F_j)$. This implies that every vertex of $F_k$ is at distance at most 2 from $r(F_k)$. Hence, the maximum distance between two vertices in $F_k$ is at most four and $F_k$ is $P_6$-free.

A graph $F_k$ has the following useful properties (see also Figure 4).

**Lemma 7** *Any graph $F_k$ with $k \geq 4$ contains copies $F_t^1$ and $F_t^2$ of $F_t$ for all $2 \leq t \leq k - 2$ such that*

(i) *For all $1 \leq i, j \leq 2$ and all $2 \leq s < t \leq k - 2$, the graphs $F_s^i$ and $F_t^j$ are remote.*

(ii) *For all $2 \leq t \leq k - 2$, the vertices of the graph $F_t^1$ in the bipartite class containing the root vertex of $F_t^1$ are adjacent to $r(F_k)$.*

(iii) *For all $2 \leq t \leq k - 2$, the vertices of the graph $F_t^2$ in the bipartite class not containing the root vertex of $F_t^2$ are adjacent to $r(F_k)$.*

*Proof.* One easily checks that child graph $F_{t+1}$ of graph $F_k$ contains the desired copies $F_t^1$ and $F_t^2$ for $2 \leq t \leq k - 2$. ∎

The following proposition is a key ingredient for the proof of our main result. It also implies that there exists no competitive on-line coloring algorithm for the family of $P_6$-free bipartite graphs (cf. [9], where this has been proven using a different subfamily of $P_6$-free bipartite graphs).

**Proposition 8** *For any $k \geq 1$, $\chi_{OL}(F_{2k}) \geq k$.*

*Proof.* By induction on $k$. The case $k = 1$ is trivial. Let $k \geq 2$. Consider $F_{2k}$ and an on-line algorithm $A$ for coloring $F_{2k}$. The first time the $i^{th}$ color is used by $A$ we identify it as color $i$. We choose an ordering on $V(F_{2k})$ such that the vertices of remote copies of $F_2, F_4, \ldots, F_{2k-2}$ are presented until color $i$ is used on $F_{2i}$ ($i = 1, \ldots, k - 1$), i.e., as soon as color 1 is used on $F_2$ we start presenting vertices of $F_4$, as soon as color 2 is used on $F_4$ we start presenting vertices of $F_6$ and so on, until color $k - 1$ is used on $F_{2k-2}$. By the adjacency relations from the definition of $F_{2k}$ and the properties of Lemma 7, the ordering of the presented vertices of $F_2, F_4, \ldots, F_{2k-2}$ can
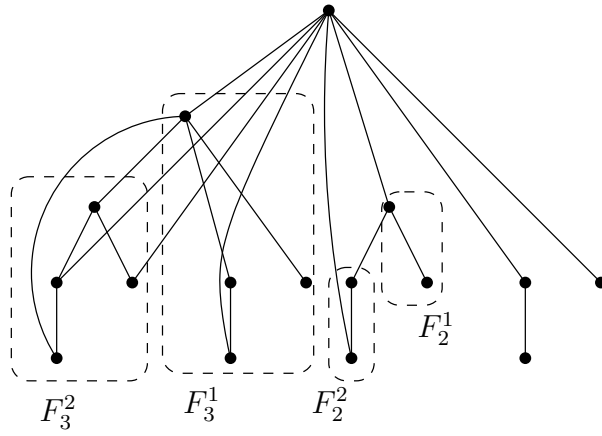
16

Figure 4: The graph $F_5$ which contains copies $F_2^1, F_2^2, F_3^1, F_3^2$.

be chosen in such a way that $r(F_{2k})$ is adjacent to the (not necessarily root) vertices that received colors $1, \ldots, k-1$. Hence a new color $k$ is forced upon $A$. ∎

Below we denote a copy of a graph $F_k$ with root vertex $v$ by $F_k(v)$. The child graphs of $F_k(v)$ are denoted by $F_1^v, F_2^v, \ldots, F_{k-1}^v$.

**Lemma 9** *Let $G$ be a $P_7$-free bipartite graph. If BicolorMax uses color $a_k$ or $b_k$ with $k \geq 3$ on vertex $v = \pi(j)$, then $C_j^{k-1}$ contains the graph $F_{\lfloor \frac{k-1}{2} \rfloor}(v)$ as an induced subgraph.*

*Proof.* Due to Lemma 6 the component $C_j^{k-1}$ contains the tree $T_{k-1}(v)$ as a subgraph in such a way that:

(i) If there exists an edge in $G$ between any two vertices $x, y$ in $T_{k-1}(v)$ with $d(v, x, T_{k-1}(v)) \leq d(v, y, T_{k-1}(v))$, then $x$ lies on the path from $y$ to $v$ in $T_{k-1}(v)$.

(ii) The root of child tree $T_i^v$ is colored with $a_{i+1}$ or $b_{i+1}$ for all $1 \leq i \leq k-2$.

By induction on $k$ we will show that the subgraph of $G$ induced by $V(T_{k-1}(v))$ contains the graph $F_{\lfloor \frac{k-1}{2} \rfloor}(v)$ as an induced subgraph. The case $k = 3$ is trivial. Let $k \geq 4$.

*Claim.* The following is true for at least $k - 3$ child trees $T_i^v$: vertex $v$ is adjacent to every vertex in $T_i^v$ that has distance 2 to $r(T_i^v)$.

We prove this claim by contradiction. Suppose there exist two child trees of $T_{k-1}(v)$ not satisfying the claim. Let $T_h^v$ be a child tree of $T_{k-1}(v)$ for some $1 \leq h \leq k-2$ that contains a vertex $x$ with $d(x, r(T_h^v), T_h^v) = 2$ and with $\{x, v\} \notin E(G)$. Let $T_j^v$ be a child tree of $T_{k-1}(v)$ for some $1 \leq j \neq h \leq k-2$ that contains a vertex $y$ with $d(y, r(T_j^v), T_j^v) = 2$ and with $\{y, v\} \notin E(G)$. Let $y'$ be the common neighbor of $y$ and $r(T_j^v)$ in $T_j^v$. Let $x'$ be the common neighbor of $x$ and $r(T_h^v)$ in $T_h^v$. Since $T_{k-1}(v)$ satisfies (i), $G$ contains an induced $P_7 = xx'r(T_h^v)vr(T_j^v)y'y$, which is a contradiction. Hence the above claim has been proven.

By the induction hypothesis and condition (ii), for $2 \leq i \leq k-2$, the subgraph induced by $V(T_i^v)$ contains the graph $F_{\lfloor \frac{i}{2} \rfloor}(r(T_i^v))$ as an induced subgraph. This, together with the above claim and the fact that $T_{k-1}(v)$ satisfies condition (i), implies that $C_j^{k-1}$ contains the graph $F_{\lfloor \frac{k-1}{2} \rfloor}(v)$ as an induced subgraph. ∎

17

**Theorem 10** *If $G$ is a $P_7$-free bipartite graph, then $\chi_{BM}(G) \leq 8\chi_{OL}(G) + 8$.*

*Proof.* Let $k$ be the highest index such that *BicolorMax* uses color $a_{4k+1}$ on a vertex in $G$. Note that it is possible that *BicolorMax* uses colors $a_{4k+2}, a_{4k+3}, a_{4k+4}$ or $b_{4k+2}, b_{4k+3}, b_{4k+4}$ to color $G$. Hence, every color used on a vertex of $G$ is from $A(4k+4) \cup B(4k+4)$. Since *BicolorMax* only uses $b_i$ if $a_i$ has been used before, $\chi_{B<}(G) \leq 2(4k+4) = 8k+8$. For $k = 0$ the statement obviously holds. Suppose $k \geq 1$. Due to Lemma 9, $G$ contains a copy of $F_{2k}$ as an induced subgraph. Proposition 8 implies that $\chi_{OL}(G) \geq \chi_{OL}(F_{2k}) \geq k$. ∎

# 7 New upper bounds on $\chi_{OL}$ for bipartite graphs

As we noted in Section 5, in [18], the authors leave it as an "easy exercise" to construct an on-line coloring algorithm $A$ such that $\chi_A(G) = O(\log_2 n)$ on the class of bipartite graphs (where $G$ is a bipartite graph on $n$ vertices). There are several choices for $A$ to solve this exercise. One of these choices is the algorithm *BicolorMax* that we presented and analyzed in this paper.

As a byproduct, using the derived properties of *BicolorMax*, below we give an upper bound on $\chi_A(G)$ for a bipartite graph $G$ in terms of the independence number $\alpha(G)$ and in terms of the number of remote induced $P_5$s in $G$. It is not possible to prove an upper bound in terms of induced subgraphs isomorphic to $P_6$, since it has been proven in [9] (and also follows from Proposition 8) that no competitive algorithm exists for the family of $P_6$-free bipartite graphs.

**Theorem 11** *Let $G$ be a bipartite graph in which each component has at most $s$ remote induced subgraphs isomorphic to $P_5$. If $s = 0$, then $\chi_{BM}(G) \leq 4$. If $s > 0$, then $\chi_{BM}(G) \leq 2\log_2(s) + 6$.*

*Proof.* We prove the theorem by showing that a component $C$ of $G$ contains at least $2^{k-3}$ remote induced subgraphs isomorphic to $P_5$, if *BicolorMax* uses color $a_k$ on $C$ with $k \geq 3$. We use induction on $k$.

Let $k = 3$. It is easy to check that a component of $G$ contains an induced $P_5$, if *BicolorMax* uses color $a_3$ on a vertex of that component. Let $k \geq 4$. Suppose $v = \pi(j)$ is colored by $a_k$ or $b_k$. Then there exists a $(k-1)$-mixed pair $(x, y)$ in $C_j^{k-1}$. By Lemma 3, $x$ and $y$ belong to two different components in $G_j^{k-1} - v$ both containing $2^{k-4}$ remote induced subgraphs isomorphic to $P_5$. ∎

Our next result gives an upper bound on the number of colors used by *BicolorMax* on a bipartite graph $G$ in terms of its independence number $\alpha(G)$ (i.e., the largest number of vertices of $G$ that are mutually nonadjacent in $G$).

**Theorem 12** *Let $G$ be a bipartite graph with independence number $\alpha$. Then*

$$\chi_{BM}(G) \leq \begin{cases} \alpha + 1 & \text{if } 1 \leq \alpha \leq 3 \\ 4 & \text{if } \alpha = 4 \\ 2p & \text{if } 5 \cdot 2^{p-3} \leq \alpha \leq 5 \cdot 2^{p-2} - 1 \text{ for some } p \geq 3. \end{cases}$$

*Proof.* Let $k$ be the highest index such that *BicolorMax* uses color $a_k$ on a vertex $v$ in $G$. If $\alpha = 1$, then $k = 1$, hence $\chi_{BM}(G) \leq 2$. If $\alpha = 2$, then $k \leq 2$ but color $b_2$ is never used, hence $\chi_{BM}(G) \leq 3$. Now assume that $\alpha \geq 3$.

Let $v = \pi(j)$. We write $C_j^{k-1} = (I_1^k, I_2^k)$ for the bipartition with $v \in I_1^k$. Clearly, $|I_1^2| \geq 2$ and $|I_2^2| \geq 2$. We will use induction to show that $|I_1^k| \geq 5 \cdot 2^{k-3}$ and $|I_2^k| \geq 5 \cdot 2^{k-3} - 1$ for $k \geq 3$.

For a fixed $k \geq 3$, by definition of *BicolorMax*, there exists a $(k-1)$-mixed pair $(x,y)$ in $C_j^{k-1}$ with $x \in I_1^k$ and $y \in I_2^k$. By Lemma 3, $x$ and $y$ belong to two different components $D = (J_1, J_2)$ and $D' = (J_1', J_2')$ in $G_j^{k-1} - v$, say $x \in J_1 \subseteq I_1^k$ and $y \in J_1' \subseteq I_2^k$. For $k = 3$, we get $|I_1^3| \geq |J_1| + |J_2'| + 1 = 2 + 2 + 1 = 5$, and $|I_2^3| \geq |J_2| + |J_1'| \geq 2 + 2 = 4$.

Now let $k \geq 4$. Then $|I_1^k| \geq |J_1| + |J_2'| + 1 = 5 \cdot 2^{k-4} + 5 \cdot 2^{k-4} - 1 + 1 = 5 \cdot 2^{k-3}$, and $|I_2^k| \geq |J_2| + |J_1'| \geq 5 \cdot 2^{k-4} - 1 + 5 \cdot 2^{k-4} = 5 \cdot 2^{k-3} - 1$. We conclude that $|I_1^k| \geq 5 \cdot 2^{k-3}$ and $|I_2^k| \geq 5 \cdot 2^{k-3} - 1$ for $k \geq 3$.

Assume $3 \leq \alpha \leq 4$. If $k = 3$, then $\alpha \geq |I_1^3| \geq 5$. Hence $k \leq 2$. Then $\chi_{BM} \leq 2k \leq 4$.

Finally assume $5 \cdot 2^{p-3} \leq \alpha \leq 5 \cdot 2^{p-2} - 1$ for some $p \geq 3$. If $k \geq p+1$, then $\alpha \geq |I_1^k| \geq 5 \cdot 2^{p-2}$. Hence $k \leq p$. Then $\chi_{BM} \leq 2p$. This completes the proof of Theorem 12. ∎

The bounds of Theorem 12 improve a bound in [5] for bipartite graphs: the main result of [5] applied to bipartite graphs shows that for any bipartite graph $G$ any Greedy on-line coloring algorithm uses at most $\alpha(G) + 1$ colors.

# 8    Final remarks

As we noted in Section 5, *BicolorMax* is not on-line competitive for the family of paths. This is in contrast to *FF*, which uses at most three colors on any path $P_n$. Below we show in a more general way how to combine two on-line competitive algorithms satisfying certain conditions.

For this purpose we need the following terminology. We call a family of graphs $\mathcal{G}$ *decidable* if there exists a (finite) procedure to decide whether or not a graph $G$ belongs to $\mathcal{G}$. Note that the family of $H$-free graphs is decidable for any fixed graph $H$. Also the family of paths is decidable. We call an on-line coloring algorithm $A$ for a graph $G$ *component-independent* if the color assigned by $A$ to every revealed vertex $v = \pi(j)$ only depends on the component of $G[\{\pi(1), \ldots, \pi(j)\}]$ that vertex $v$ belongs to. For instance, *FF* and *BicolorMax* are component-independent. For two families of graphs $\mathcal{F}$ and $\mathcal{G}$ we define $\mathcal{F} + \mathcal{G} = \{L \mid L \in \mathcal{F} \cup \mathcal{G}, \text{ or } L = F + G \text{ with } F \in \mathcal{F} \text{ and } G \in \mathcal{G}\}$.

**Theorem 13** *Let $\mathcal{F}$ be a decidable family of graphs for which an on-line competitive coloring algorithm $A_1$ exists, and let $\mathcal{G}$ be a decidable family of graphs for which an on-line competitive component-independent coloring algorithm $A_2$ exists. Then there exists an on-line coloring algorithm $A_3$ that is on-line competitive for $\mathcal{F} + \mathcal{G}$.*

*Proof.* We assume $A_1$ and $A_2$ use disjoint color sets $\{c_1, c_2, \ldots\}$ an $\{d_1, d_2, \ldots\}$ respectively. Let $\pi \in \Pi(L)$ for some graph $L \in \mathcal{F} + \mathcal{G}$.

We inductively define $A_3$ as follows. The vertex $\pi(1)$ is colored with $d_1$. Suppose that vertices $\pi(1), \ldots, \pi(j-1)$ have already been colored and let $v = \pi(j)$ be the next vertex of $L$ presented to $A_3$. For $i = 1, 2$, we let $L_j(A_i)$ be the subgraph of $L$ that consists of $v$ and all vertices in $\{\pi(1), \ldots, \pi(j-1)\}$ colored by $A_i$. We let $C_j(A_2)$ be the component of $L_j(A_2)$ that contains $v$. If $C_j(A_2)$ is an induced subgraph of some graph in $\mathcal{G}$ then color $v$ with the color that $A_2$ would have used on $v$ after the other vertices of $C_j(A_2)$ had been presented to $A_2$ in the suborder defined by $\pi$. Otherwise $L_j(A_1)$ is a subgraph of a graph in $\mathcal{F}$, and we color $v$ with the color that $A_1$ would have used on $v$ after the other vertices of $L_j(A_1)$ had been presented to $A_1$ in the suborder defined by $\pi$. It is easy to see that $A_3$ is on-line competitive on $\mathcal{F} + \mathcal{G}$. ∎

# 9 Conclusions and future work

We have presented an on-line coloring algorithm *BicolorMax* for bipartite graphs. We have shown that the number of colors used by this algorithm on a bipartite graph $G$ is bounded from above by the number of remote induced subgraphs of $G$ isomorphic to $P_5$ and we gave a similar upper bound in terms of the independence number of $G$. As a consequence we improved known upper bounds for the on-line chromatic number of bipartite graphs given in [5, 18]. We showed in [3] that for any $P_6$-free bipartite graph $G$, *BicolorMax* uses at most twice as many colors as any optimal on-line coloring algorithm for $G$. Here we showed that for any $P_7$-free bipartite graph $G$, *BicolorMax* uses at most eight times as many colors as any optimal on-line coloring algorithm for $G$.

In a future continuation of this work, we would like to face the problem of deciding whether for any $n \geq 8$, a linear on-line competitive algorithm can be defined for the class of $P_n$-free bipartite graphs. We also consider analyzing *BicolorMax* and related algorithms for other classes of $H$-free bipartite graphs, in particular for bipartite graphs $H$ with 6 vertices. A seemingly difficult and interesting open case is the (non)existence of an on-line competitive algorithm for the class of $C_6$-free bipartite graphs.

# Acknowledgments

# References

[1] J.A. BONDY AND U.S.R. MURTY, *Graph Theory with Applications,* Macmillan, London and Elsevier, New York (1976).

[2] A. BORODIN AND R. EL-YANIV, *Online Computation and Competitive Analysis,* Cambridge University Press (1998).

[3] H.J. BROERSMA, A. CAPPONI, AND D. PAULUSMA, On-line coloring of H-free bipartite graphs. In: *Proceedings of the 6th Italian Conference on Algorithms and Complexity (CIAC 2006)* (2006), Lecture Notes in Computer Science 3998, Springer Verlag, 284–295.

[4] M. CHROBAK AND M. ŚLUSAREK, Problem 84-23. *Journal of Algorithms 5* (1984), 588.

[5] I. CIEŚLIK, M. KOZIK, AND P. MICEK, On-line coloring of $I_s$-free graphs and co-planar graphs. *Discrete Mathematics and Theoretical Computer Science Proc. AF* (2006), 61–68.

[6] T. ERLEBACH AND J. FIALA, On-line coloring of geometric intersection graphs. *Computational Geometry: Theory and Applications 23* (2002), 243–255.

[7] A. GYÁRFÁS, Z. KIRÁLY, AND J. LEHEL, On-line competitive coloring algorithms. *Technical report TR-9703-1* (1997), available at http://www.cs.elte.hu/tr97/tr9703-1.ps.

[8] A. GYÁRFÁS, Z. KIRÁLY, AND J. LEHEL, On-line 3-chromatic graphs. II. Critical graphs. *Discrete Mathematics 177* (1997), 99–122.

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

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

[11] A. Gyárfás and J. Lehel, Effective on-line coloring of $P_5$-free graphs. *Combinatorica 11* (1991), 181–184.

[12] M.M. Halldórsson and M. Szegedy, Lower bounds for on-line graph coloring. *Theoretical Computer Science 130* (1994), 163–174.

[13] H.A. Kierstead, The linearity of first-fit coloring of interval graphs. *SIAM Journal on Discrete Mathematics 1* (1988), 526–530.

[14] H.A. Kierstead, Coloring graphs on-line. In: *Online algorithms: the state of the art* (1998), Lecture Notes in Computer Science 1442, Springer Verlag, 281–305.

[15] H. Kierstead and S.G. Penrice Radius two trees specify $\chi$-bounded classes. *Journal of Graph Theory 18* 1994, 119–129.

[16] H. Kierstead, S.G. Penrice, and W. Trotter, On-line graph coloring and recursive graph theory. *SIAM Journal on Discrete Mathematics 7* 1994, 72–89.

[17] K. Kolossa, On the on-line chromatic number of the family of on-line 3-chromatic graphs. *Discrete Mathematics 150* (1996), 205–230.

[18] L. Lovász, M. Saks, and W.T. Trotter, An on-line graph coloring algorithm with sublinear performance ratio. *Discrete Mathematics 75* (1989), 319–325.

[19] B. Randerath and I. Schiermeyer, Vertex colouring and forbidden subgraphs – a survey. *Graphs and Combinatorics 20* (2004), 1–40.

[20] D.D. Sleator and R.E. Tarjan, Amortized efficiency of list updates and paging rules. *Comm. ACM 28* (1985), 202–208.