IMFM
Institute of Mathematics, Physics and Mechanics
Jadranska 19, 1000 Ljubljana, Slovenia

Preprint series
Vol. 48 (2010), 1111
ISSN 2232-2094

A PLANAR LINEAR ARBORICITY CONJECTURE

Marek Cygan Łukasz Kowalik
Borut Lužar

Ljubljana, January 19, 2010

A Planar Linear Arboricity Conjecture*

Marek Cygan[1], Łukasz Kowalik[1], Borut Lužar[2†]

[1] Institute of Informatics, Warsaw University,
Banacha 2, 02-097 Warsaw, Poland
{cygan,kowalik}@mimuw.edu.pl
[2] Institute of Mathematics, Physics, and Mechanics,
Jadranska 19, 1111 Ljubljana, Slovenia
borut.luzar@gmail.com

Abstract

The linear arboricity $\mathrm{la}(G)$ of a graph G is the minimum number of linear forests (graphs where every connected component is a path) that partition the edges of G. In 1984, Akiyama et al. [1] stated the Linear Arboricity Conjecture (LAC), that the linear arboricity of any simple graph of maximum degree Δ is either $\lceil \frac{\Delta}{2} \rceil$ or $\lceil \frac{\Delta+1}{2} \rceil$. In [14, 17] it was proven that LAC holds for all planar graphs.

LAC implies that for Δ odd, $\mathrm{la}(G) = \lceil \frac{\Delta}{2} \rceil$. We conjecture that for planar graphs this equality is true also for any even $\Delta \geq 6$. In this paper we show that it is true for any even $\Delta \geq 10$, leaving open only the cases $\Delta = 6, 8$.

We present also an $O(n \log n)$ algorithm for partitioning a planar graph into $\max\{\mathrm{la}(G), 5\}$ linear forests, which is optimal when $\Delta \geq 9$.

1 Introduction

In this paper we consider only undirected and simple graphs. A *linear forest* is a forest in which every connected component is a path. The linear arboricity $\mathrm{la}(G)$ of a graph G is the minimum number of linear forests in G, whose union is the set of all edges of G. This one of the most natural graph covering notions was introduced by Harary [9] in 1970. Note that for any graph of maximum degree Δ one needs at least $\lceil \frac{\Delta}{2} \rceil$ linear forests to cover all the edges. If Δ is even and the graph is regular, $\lceil \frac{\Delta}{2} \rceil$ forests do not suffice, for otherwise every vertex in every forest has degree 2, a contradiction. Hence, for any Δ-regular graph G, we have $\mathrm{la}(G) \geq \lceil \frac{\Delta+1}{2} \rceil$. Akiyama, Exoo and Harary conjectured that this bound is always tight, i.e. for any regular graph, $\mathrm{la}(G) = \lceil \frac{\Delta+1}{2} \rceil$. It is easy to see (check e.g. [3]) that this conjecture is equivalent to

Conjecture 1. *For any graph G, $\lceil \frac{\Delta}{2} \rceil \leq \mathrm{la}(G) \leq \lceil \frac{\Delta+1}{2} \rceil$.*

We note that Conjecture 1 resembles Vizing Theorem, and indeed, for odd Δ it can be treated as a generalization of Vizing Theorem (just color each linear forest with two new

*Supported in part by bilateral project BI-PL/08-09-008. M. Cygan and Ł. Kowalik were supported in part by Polish Ministry of Science and Higher Education grant N206 355636

†Operation part financed by the European Union, European Social Fund.

Preprint series, IMFM, ISSN 2232-2094, no. 1111, January 19, 2010

colors and get a $(\Delta+1)$-edge-coloring). However, despite many efforts, the conjecture is still open and the best known general upper bound is $\mathrm{la}(G) = \frac{d}{2} + O(d^{2/3}(\log d)^{1/3})$, due to Alon, Teague and Wormald [4]. Conjecture 1 was proved only in several special cases: for $\Delta = 3, 4$ in [1, 2], for $\Delta = 5, 6, 8$ in [8], $\Delta = 10$ [7], for bipartite graphs [1] to mention a few. Finally, it was shown for planar graphs (for $\Delta \neq 7$ by Jian-Liang Wu [14] and for $\Delta = 7$ by Jian-Liang Wu and Yu-Wen Wu).

In this paper we focus on planar graphs. Note that for all odd Δ, we have $\lceil\frac{\Delta}{2}\rceil = \lceil\frac{\Delta+1}{2}\rceil$. Hence, the linear arboricity is $\lceil\frac{\Delta}{2}\rceil$ for planar graphs of odd Δ. Moreover, in [14], Wu showed that this is also true for $\Delta \geq 13$. In this paper we state the following conjecture.

Conjecture 2. *For any planar graph G of maximum degree $\Delta \geq 6$, we have* $\mathrm{la}(G) = \lceil\frac{\Delta}{2}\rceil$.

It is easy to see that the above equality does not hold for $\Delta = 2, 4$, since there are 2- and 4-regular planar graphs: e.g. the 3-cycle and the octahedron. Interestingly, if the equality holds for $\Delta = 6$, then any planar graph with maximum degree 6 is 6-edge-colorable (just edge-color each of the three linear forests in two new colors). Hence, Conjecture 2 implies the Vizing Planar Graph Conjecture [13] (as currently it is open only for $\Delta = 6$):

Conjecture 3 (Vizing Planar Graph Conjecture). *For any planar graph G of maximum degree $\Delta \geq 6$, we have $\chi'(G) = \Delta$.*

The main result of this paper is the following theorem.

Theorem 1. *For every planar graph G of maximum degree $\Delta(G) \geq 9$, we have* $\mathrm{la}(G) = \lceil\frac{\Delta}{2}\rceil$.

We note that Wu, Hou and Sun [16] verified Conjecture 2 for planar graphs without 4-cycles. For $\Delta \geq 7$ it is also known to be true for planar graphs without 3-cycles [14] and without 5-cycles [15].

Computational Complexity Perspective. Consider the following decision problem. Given graph G and a number k, determine whether $\mathrm{la}(G) = k$. Peroche [11] showed that this problem is NP-complete even for graphs of maximum degree $\Delta = 4$. Our result settles the complexity of the decision problem for planar graphs of maximum degree $\Delta(G) \geq 9$. The discussion above implies that for planar graphs the decision problem is trivial also when Δ is odd. When $\Delta = 2$ the problem is in P, as then the algorithm just checks whether there is a cycle in G. Hence, the remaining cases are $\Delta = 4, 6, 8$. Conjecture 2, if true, excludes also cases $\Delta = 6, 8$. We conjecture that the only remaining case $\Delta = 4$ is NP-complete.

Conjecture 4. *It is NP-complete to determine whether a given planar graph of maximum degree 4 has linear arboricity 2.*

Finally, even when one knows the linear arboricity of a given graph (or a bound on it) the question arises how fast one can find the corresponding collection of linear forests. We show the following result.

Theorem 2. *For every n-vertex planar graph G of maximum degree Δ, one can find a cover of its edges by $\max\{\mathrm{la}(G), 5\}$ linear forests in $O(n \log n)$ time.*

Preprint series, IMFM, ISSN 2232-2094, no. 1111, January 19, 2010

Preliminaries. For any graph G, $V(G)$ and $E(G)$ denote the sets of vertices and edges of G, respectively. If G is plane, we also denote the set of its faces by $F(G)$.

We call a vertex of degree k, at least k, and at most k, a k-vertex, $(\geq k)$-vertex, and $(\leq k)$-vertex, respectively. Moreover, a neighbor u of a vertex v is called a k-neighbor, where $k = \deg(u)$. In a plane graph, *length* of a face f, denoted as $\ell(f)$ is the number of edges incident to f. Analogously to vertices, a face of length k, at least k, and at most k, is called a k-face, $(\geq k)$-face, and $(\leq k)$-face, respectively. By a *weight* of an edge uv we denote the sum of degrees of its endpoints.

Throughout the paper it will be convenient for us to treat partitions into linear forests as a kind of edge-colorings. A *k-linear coloring* of a graph $G = (V, E)$ is a function $C : E \to \{1, \ldots, k\}$ such that for $i = 1, \ldots, k$, the set of edges $C^{-1}(i)$ is a linear forest. We call an edge colored by a, an a-edge.

Following the notation in [14], by $C_0(v)$, we denote the set of colors that are not used on edges incident to the vertex v, and $C_1(v)$ is the set of colors which are assigned to exactly one edge incident to v. Finally, $C_2(v) = \{1, \ldots, k\} \setminus (C_0(v) \cup C_1(v))$. We call the color which is in $C_0(v) \cup C_1(v)$ a *free* color at v. A path, where all edges have the same color a is called a *monochromatic path* or an *a-path*. A *(uv, a)-path* is an a-path with endpoints u and v.

We use also the Iverson's notation, i.e. $[\alpha]$ is 1 when α is true and 0 otherwise.

2 Proof of Theorem 1

It will be convenient for us to prove the following generalization of Theorem 1:

Proposition 3. *Any simple planar graph of maximum degree* Δ *has a linear coloring in* $\max\{\lceil \frac{\Delta}{2} \rceil, 5\}$ *colors.*

Our plan for proving Proposition 3 is as follows. First we are going to show some structural properties of a graph which is *not* k-linear colorable and, subject to this condition, is minimal, i.e. has as few edges as possible. In this phase we do not use planarity, and our results apply to general graphs. Next, using the so-called discharging method (a powerful technique developed for proving the four color theorem) we will show that when $k = \max\{\lceil \frac{\Delta}{2} \rceil, 5\}$, there is no planar graph with the obtained structural properties.

2.1 Structure of a minimal graph G with $\Delta(G) \leq 2k$ and $\mathrm{la}(G) > k$

In this section we fix a number k and we assume that G is a graph of maximum degree at most $2k$ which is *not* k-linear colorable and, among such graphs, G has minimal number of edges. The following Lemma appears in many previous works, e.g. in [15], but we give the proof for completeness.

Lemma 4. *For every edge* uv *of* G, $d(u) + d(v) \geq 2k + 2$.

Proof. Suppose for a contradiction that uv is an edge in G, and $d(u) + d(v) < 2k + 2$. By the minimality of G, there exists a k-linear coloring of $G' = G - uv$. Note that the degree of vertices u and v in G' is one less than in G, hence $d_{G'}(u) + d_{G'}(v) < 2k$. So there exists at least one color c which is either an element of $C_0(u)$ and free at v, or an element of $C_0(v)$ and free at u. It follows that we can extend the k-linear coloring of G' to G, a contradiction. $\square$

Preprint series, IMFM, ISSN 2232-2094, no. 1111, January 19, 2010

Lemma 5. *G does not contain a 2-vertex v such that the two neighbors u and z are not adjacent.*

Proof. Since there is no edge uz we can create a simple graph $G' = (G - v) \cup uz$. Because of the minimality of G there exists a k-linear coloring C of G'. Let $a = C(uz)$. We can put $C(uv) = C(vz) = a$ obtaining a k-linear coloring of G, a contradiction. □

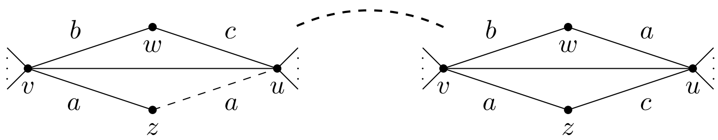
Lemma 6. *Every vertex has at most one adjacent 2-vertex.*

(A) 2-vertices have two common neighbors (B) 2-vertices have only one common neighbor

Figure 1: Cases A and B in the proof of Lemma 6

Proof. Assume for a contradiction that a vertex u has at least two neighbors of degree two namely w and z. By Lemma 5, neighbors of every 2-vertex are adjacent. Consider the configuration and the labeling of vertices as in Fig. 1.

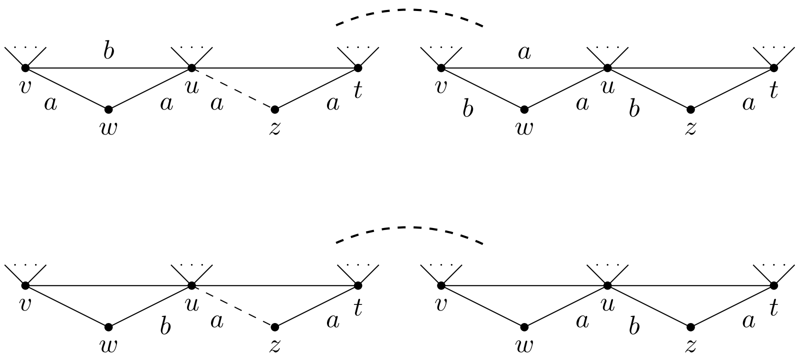
Case A. The vertices w and z have two common neighbors. Since G is minimal, there exists a k-linear coloring C of $G' = G - uz$. Let a be a free color at u. We may assume that $C(vz) = a$ and there exists a (uv, a)-path, otherwise we can put $C(uz) = a$ and we are done. Consider edges vw and wu. If at least one of them has color a it means that both have color a since there is a (uv, a)-path. In such a case ($C(vw) = C(wu) = a$) we can take color $b = C(uv) \neq a$ and recolor $C(vw) = C(wu) = b$ and $C(uv) = a$. Thus we can assume that $C(vw) = b \neq a$ and $C(wu) = c \neq a$ therefore it suffices to recolor $C(uw) = a$ and put $C(uz) = c$.



Case B. Vertices w and z have precisely one common neighbor. Since G is minimal, there exists a k-linear coloring C of $G' = G - uz$. Let a be the only free color at u. We may assume that $C(tz) = a$ and there exists a (tu, a)-path since otherwise we can put $C(uz) = a$ and we are done. Now let us take into consideration the edge uw.

Case B.1. $C(uw) = a$. Since in G' the only a-edge incident to the vertex u is the edge uw we have $C(wv) = a$ because there exists a (tu, a)-path in G'. Let $C(uv) = b$. We know that $b \neq a$ since a is free at u. In this case we recolor $C(vw) = b$, $C(uv) = a$ and $C(uz) = b$. It is easy to see that we do not introduce any monochromatic cycle.

Case B.2. $C(uw) = b \neq a$. In this case we only recolor $C(uw) = a$ and put $C(uz) = b$. Even if $C(vw) = a$ we do not introduce an a-cycle since then a (uv, a)-path can not exist because of the (tu, a)-path.



□

Lemma 7. *G does not contain a* 3*-vertex with precisely two pairs of adjacent neighbors.*

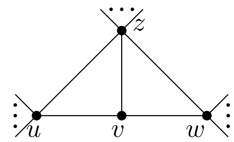


Figure 2: A 3-vertex with precisely two pairs of adjacent neighbors.

Proof. Assume the configuration described in the claim exists and consider the labeling of vertices as in Fig. 2, in particular $uw \notin E(G)$. Let G' be the graph obtained by removing v and adding the edge uw to G. By the minimality of G, there exists a k-linear coloring C of G'. Let a, b, and c be the colors of the edges uz, zw, and uw, respectively. We show that C can be extended to G as follows.

First, color the edges in $E(G) \cap E(G')$ as in the coloring C. Then only the edges uv, vw, and vz remain non-colored. Let d be a color free at z. If $d \neq c$ we color vz by d and both vu, vw by c. It is easy to see that this is a proper k-linear coloring of G (note that there is no (uw, c)-path in G, since then there is a c-cycle in the coloring C). So, we may assume that c is free at z. Now observe that if in the partial coloring of G there are both a (uz, c)-path and (wz, c)-path then G' contains a c-cycle. By symmetry, we can assume that there is no (uz, c)-path. Then, in particular, $c \neq a$. We color the edges uv and vz with a, vw, uz with c, and wz remains colored with b. Thus we obtained a k-linear coloring of G, a contradiction. □

Lemma 8. *G does not contain the configuration of Fig. 3, i.e. a 4-cycle $vzuw$ with a chord zw such that* $\deg(v) = 3$ *and* $\deg(u) = 2$.

Proof. Consider the configuration and the labeling of vertices as in Fig. 3. Since G is a minimal counterexample, there exists a k-linear coloring C of $G' = G - uz$. Let a be a free color at z. We may assume that $C(uw) = a$ and there exists a (wz, a)-path since otherwise we can put $C(uz) = a$ and we are done. Now let us take into consideration the edge wz.

Case 1. $C(wz) = a$. then since a is free at the vertex v we know that $C(vz) = b \neq a$ and $C(vw) = c \neq a$ since the vertex w already has two incident edges of the color a. In this case we recolor $C(vz) = a$ and put $C(uz) = b$ and we do not introduce any monochromatic cycle.

Preprint series, IMFM, ISSN 2232-2094, no. 1111, January 19, 2010

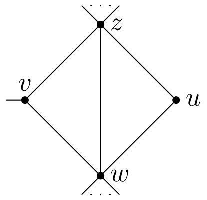


Figure 3: Configuration from Lemma 8

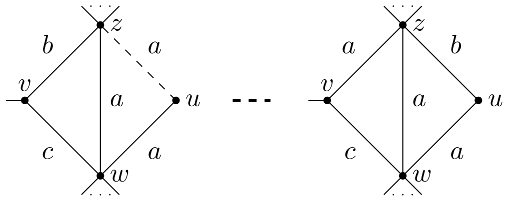


Figure 4: Case 1 of the proof of Lemma 8.

Case 2. $C(wz) = b \neq a$. Now we consider the edges vz and vw:

Case 2.1 $C(wz) = b \neq a$, $C(vz) = c \neq a$, $C(vw) = d \neq a$. In this case we recolor $C(vz) = a$ and put $C(uz) = c$. We do not introduce any a-cycle because there is a (wz,a)-path which means that there is no (vw,a)-path since w has only one adjacent a-edge outside of the configuration.

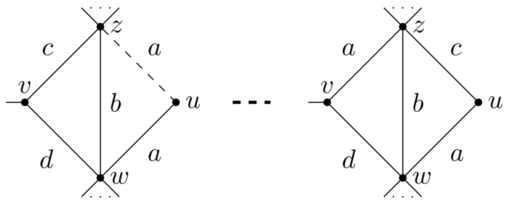


Figure 5: Case 2.1 of the proof of Lemma 8.

Case 2.2 $C(wz) = b \neq a$, $C(vz) = c \neq a$, $C(vw) = a$. In this case we recolor $C(wz) = a$, $C(vw) = b$ and put $C(uz) = b$. Since there was a (wz,a)-path it means that the only outside edge of the vertex v has color a thus even if $b = c$ we do not introduce any monochromatic cycle.

Case 2.3 $C(wz) = b \neq a$, $C(vz) = a$, $C(vw) = c \neq a$. Just note that if we uncolor wu, color zu with a and swap the names of vertices z and w we arrive at Case 2.2.

Case 2.4 $C(wz) = b \neq a$, $C(vz) = a$, $C(vw) = a$. Since there is only one outside edge incident to the vertex v there can not be simultaneously (vz, b)-path and (vw, b)-path. Because of the symmetry we may assume w.l.o.g. that there is no (vw, b)-path. In this case we recolor $C(wz) = a$,$C(vw) = b$ and put $C(uz) = b$.

□

Lemma 9. *G does not contain the configuration in Fig. 8.*

Preprint series, IMFM, ISSN 2232-2094, no. 1111, January 19, 2010

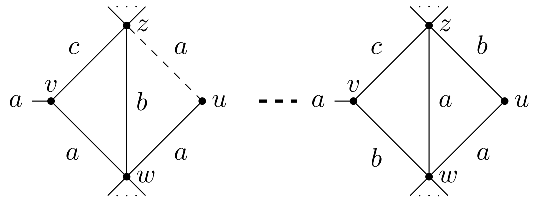


Figure 6: Case 2.2 of the proof of Lemma 8.

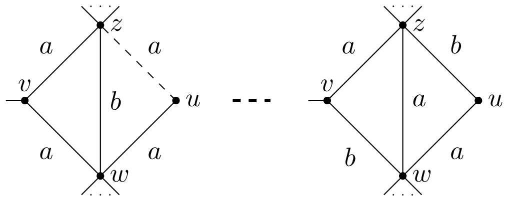


Figure 7: Case 2.4 of the proof of Lemma 8.

Proof. Consider the configuration and the labeling of vertices as in Fig. 8. By the minimality of G, there exists a k-linear coloring C of $G - uv$. We show how to extend C to G and obtain a contradiction with the minimality. The only non-colored edge is uv. Let a be the free color at v. We may assume that $a = C(uw)$ and that there exists (vu, a)-path, for otherwise we can color uv with a without introducing a monochromatic cycle.

Case 1. $C(vz) \neq a$. We color uv with $C(vz)$ and we uncolor vz, obtaining a k-linear coloring of $G - vz$ with a (vu, a)-path. We can assume that at least one of zx, zy is not colored with a, for otherwise we just recolor both zx and zy to $C(xy)$ and xy to a and we obtain another k-linear coloring of $G - vz$ with none of zx, zy colored with a. Hence, we can color vz with a and we do not introduce a monochromatic cycle because vz is on an a-path which ends at u.

Case 2. $C(vz) = a$. Since there is a (vu, a)-path, $C(zx) = a$ or $C(zy) = a$. W.l.o.g. assume $C(zx) = a$. Let $C(zy) = b$. We can assume that $C(xv) = b$ for otherwise we color both zx, vz with $C(xv)$ and xv with a and we arrive at Case 1. Let $c = C(vw)$. If $c \neq b$, or $c = b$ and every (xy, b)-path goes through the edge xv, we can color both uw and vz with c, and both vw, uv with a. Hence, we assume that $C(vw) = b$ and there is an (xy, b)-path which does not go through the edge xv. Let $d = C(vy)$. Note that a, b, d are pairwise distinct. Then we recolor the edges as in Fig. 9, that is vx with a, vy and xz with b, and uv, yz with d. We do not introduce any d-cycle, because $d \in C_1(u)$ and $d \in C_1(z)$. We do not introduce any a-cycle, because both vx and vz are on an a-path which ends at z. Finally, we do not introduce any b-cycle, because there is a (xy, b)-path so vy is on a b-path which ends at z. □

From Lemmas 7, 8 and 9 we immediately obtain the following corollary.

Corollary 10. *If a vertex v belongs to a triangle with a 2-vertex then every 3-vertex adjacent to v belongs to at most one triangle.*

Lemma 11. *If $k \geq 3$ and G contains a vertex v of degree at most $2k-1$ with two 3-neighbors then the neighbors of any 3-neighbor of v are pairwise nonadjacent.*

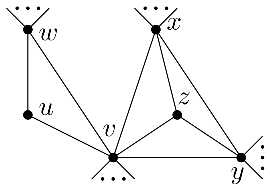


Figure 8: The configuration from Lemma 9. $\deg(u) = 2$, $\deg(z) = 3$.

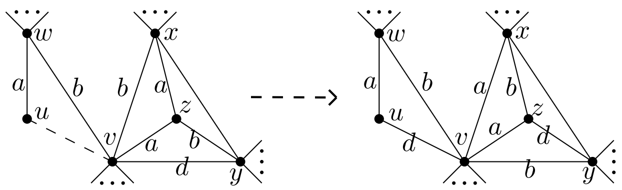


Figure 9: Proof of Lemma 9, Case 2.

Proof. Let x and y be two 3-neighbors of v and assume that a pair of neighbors of x or y, say of x, is adjacent. Let x_1, x_2 (resp. y_1, y_2) be the neighbors of x (resp. y) distinct from v. By Lemma 4 x and y are not adjacent.

Let $G' = G - vx$. Since G is minimal, there exists a k-coloring C of G'. In what follows, we show that C can be extended to G. Color the edges of G with the colors assigned in G'. The only non-colored edge is vx.

Case 1. $C_0(v) \neq \emptyset$. Let a be an element of $C_0(v)$. We immediately infer that $C(xx_1) = C(xx_2) = a$, otherwise we color vx with a and introduce no monochromatic cycle, since no other edge incident to v is colored by a. If x_1 is adjacent to v, we color xx_1 and vx with $C(x_1v)$ and x_1v with a and we obtain a k-linear coloring of G. We proceed similarly when x_2 is adjacent to v. Finally, when x_1 is adjacent to x_2, we just color xx_1 and xx_2 with $C(x_1x_2)$, and both vx and x_1x_2 with a.

Case 2. $C_0(v) = \emptyset$. Then $|C_1(v)| \geq 2$. Let a, b be two distinct elements of $C_1(v)$. Observe that if $C(xx_1)$ and $C(xx_2)$ are both distinct from one of a or b, we color vx with that color. Hence, we may assume that, without loss of generality, $C(xx_1) = a$ and $C(xx_2) = b$. There exist also (vx, a)- and (vx, b)-paths, otherwise we color vx with a or b without introducing a monochromatic cycle.

Let $c = C(vy)$. We color vx with c. Next, we color vy with a if $a \notin C_2(y)$ or with b otherwise. It is easy to check that each color induces a graph of maximum degree 2. It suffices to check that neither vx nor vy belong to a monochromatic cycle. If $c = a$, then, since there is a (vx, a)-path, $C(yy_1) = a$ or $C(yy_2) = a$, so we colored vy with b. Hence, vx is on an a-path ending at v and since there is a (vx, b)-path, vy is on a b-path ending at x. Now assume $c = b$. If we colored vy with b it means that $C(yy_1) = C(yy_2) = a$, so both vx and vy are on a b-path which ends at y. Otherwise, vx is on a b-path ending at v and since there is a (vx, a)-path, vy is on a a-path ending at x. Finally, if $c \notin \{a, b\}$, edge vx is on a c-path ending at x and edge vy is on a monochromatic path ending at x. □

Lemma 12. *G does not contain the configuration in Fig. 10.*

Proof. Consider the configuration and the labeling of vertices as in Fig. 10. Since G is minimal, there exists a k-linear coloring C of $G' = G - uv$. Let a be a free color at u. Now

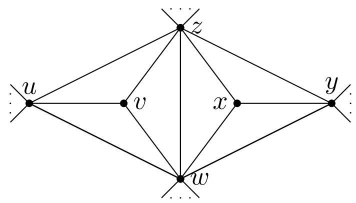


Figure 10: The configuration from Lemma 12. $\deg(v) = \deg(x) = 3$.

let us take into consideration edges vz and vw.

Case 1 $C(vz) \neq a, C(vw) \neq a$. In this case we simply put $C(uv) = a$ and we are done.

Case 2 $C(vz) = a, C(vw) = a$. Let $b = C(zw)$. Obviously $b \neq a$ so we can recolor $C(vz) = C(vw) = b$, $C(zw) = a$ and put $C(uv) = a$.

Case 3 Exactly one edge from the set $\{vz, vw\}$ has color a. Because of the symmetry we may assume that $C(vw) = a$ and $C(vz) = b \neq a$. We additionally assume that there is a (uw, a)-path since otherwise we can put $C(uv) = a$ without introducing a monochromatic cycle. Le us consider the edge wx.

Case 3.1 $C(vw) = a, C(vz) = b \neq a, C(wx) = c \neq a$ and there exists a (uw, a)-path (see Figure 11). We would like to swap colors on edges wv and wx thus in order to do so we consider subcases regarding the number of a-edges incident to the vertex x.

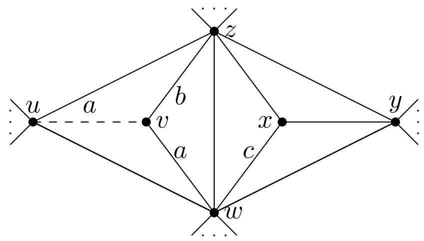


Figure 11: Case 3.1 in the proof of Lemma 12

Case 3.1.1 $C(vw) = a, C(vz) = b \neq a, C(wx) = c \neq a$, there exists a (uw, a)-path and there is at most one a-edge incident to the vertex x. Let us swap colors of edges vw and wx as in Figure 12. We know that the a-path that starts in the vertex u reaches the vertex w which means that it ends in the vertex v. Since we have an assumption that there is at most one a-edge incident to the vertex x it can not happen that the (uw, a)-path goes through the vertex x thus even if we connect those paths by swapping colors of edges vw and wx we do not introduce an a-cycle. We can only introduce a monochromatic cycle when $b = c$ and there exists a (zw, b)-path which does not go through the edge wx. We assume that this is the case since otherwise we are done. Let us consider the edge xz.

Case 3.1.1.1 $C(vw) = a, C(vz) = b \neq a, C(wx) = b, C(xz) = b$, there is at most one a-edge incident to the vertex x, there exists a (uw, a)-path and there exists a (zw, b)-path which does not go through the edge wx. Since the vertex z has already two incident b-edges the last condition can not be satisfied, contradiction (see Figure 13).

Case 3.1.1.2 $C(vw) = a, C(vz) = b \neq a, C(wx) = b, C(xz) = c \neq b$, there is at most one a-edge incident to the vertex x, there exists a (uw, a)-path and there exists a (zw, b)-path which does not go through the edge wx. In this case we swap colors of two pairs of edges

Preprint series, IMFM, ISSN 2232-2094, no. 1111, January 19, 2010

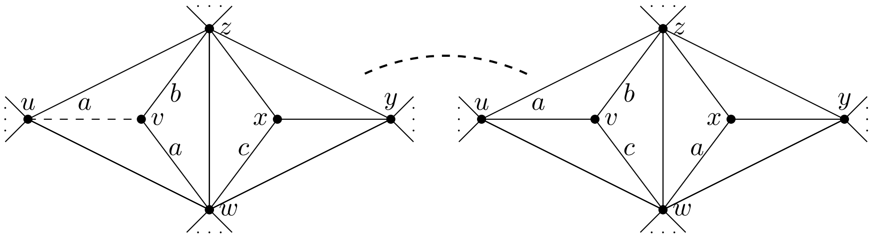


Figure 12: Case 3.1.1 in the proof of Lemma 12

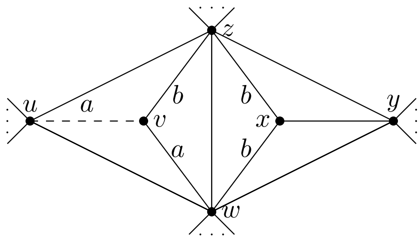


Figure 13: Case 3.1.1.1 in the proof of Lemma 12

$\{zv, zx\}$ and $\{wv, wx\}$ as in Figure 14. We show that no monochromatic cycle is introduced. Since there is a (zw, b)-path that does not go through wx, edges zx and vw are on the same b-path, which ends at v. If $c = a$ and there is an a-cycle, it means that $C(xy) = a$ and there is a (yz, a)-path which does not go through x — but then there is an a-cycle in G', a contradiction. If $c \neq a$, vz is on a c-path that ends at v and both uv and wx are on the same a-path, which ends at v.

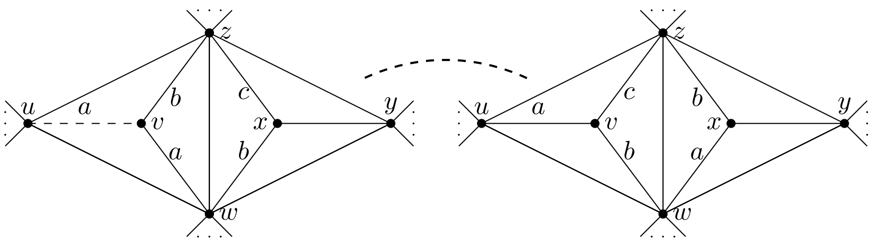


Figure 14: Case 3.1.1.2 in the proof of Lemma 12

Case 3.1.2 $C(vw) = a, C(vz) = b \neq a, C(wx) = c \neq a, C(zx) = C(xy) = a$ and there exists a (uw, a)-path. We may assume that $C(zy) = c$ since if $C(zy) = d \neq c$ we can recolor $C(zx) = C(xy) = d$ and $C(zy) = a$ which would result in the same situation as in the already solved Case 3.1.1. Now we try to recolor $C(vw) = c, C(wx) = a, C(zy) = a, C(zx) = C(xy) = c$ as in Figure 15. Now let us consider cases:

Case 3.1.2.1 We do not introduce a monochromatic cycle and we are done.

Case 3.1.2.2 We introduce a monochromatic cycle which means that $b = c$ and there exists a (wy, b)-path, hence the set of assumptions is $C(vw) = a, C(vz) = b \neq a, C(wx) = b, C(zx) = C(xy) = a, C(zy) = b$, there exists a (uw, a)-path and there exists a (wy, b)-path. In this case we consider the (uw, a)-path and branch on the relation between this path and

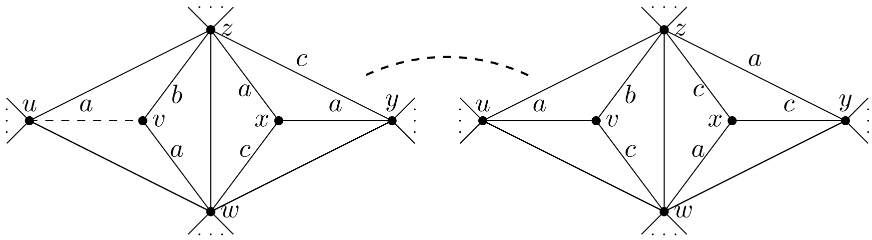


Figure 15: Case 3.1.2 in the proof of Lemma 12

vertices z, x, and y.

Case 3.1.2.2.1 $C(vw) = a, C(vz) = b \neq a, C(wx) = b, C(zx) = C(xy) = a, C(zy) = b$, there exists a (uw, a)-path, there exists a (wy, b)-path and the (uw, a)-path does not go through the vertex x. In this case we swap colors on two pairs of edges $\{zv, zx\}$ and $\{wv, wx\}$ as in Figure 16 and in this way we do not introduce any monochromatic cycle since we join distinct a-paths.

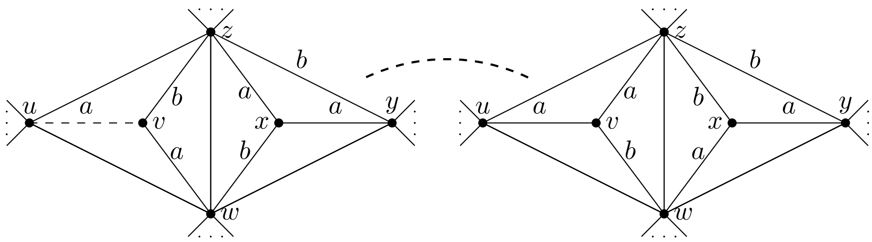


Figure 16: Case 3.1.2.2.1 in the proof of Lemma 12

Case 3.1.2.2.2 $C(vw) = a, C(vz) = b \neq a, C(wx) = b, C(zx) = C(xy) = a, C(zy) = b$, there exists a (wy, b)-path, there exists a (wz, a)-path which does not go through the vertex x and there exists a (yu, a)-path which does not go through the vertex x. Let $c = C(uz)$. Since the vertex z already has two incident b-edges we have $c \neq b$. Moreover the (yu, a)-path does not go through the vertex x thus $c \neq a$. In this case we can recolor $C(uz) = a, C(uv) = c, C(zx) = c$ as in Figure 17 and we do not introduce any monochromatic cycle because colors a, b, c are pairwise distinct and all the a-edges of the configuration are on an a-path which ends at v and x.

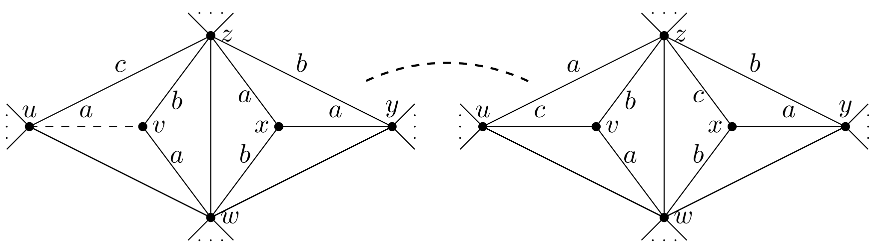


Figure 17: Case 3.1.2.2.2 in the proof of Lemma 12

Case 3.1.2.2.3 $C(vw) = a, C(vz) = b \neq a, C(wx) = b, C(zx) = C(xy) = a, C(zy) = b$, there exists a (wy, b)-path, there exists a (wy, a)-path which does not go through the vertex x and there exists a (zu, a)-path which does not go through the vertex x. Let $c = C(uw)$. Because of the (zu, a)-path we have $c \neq a$ thus we branch into cases were $c \neq b$ and $c = b$.

Case 3.1.2.2.3.1 $C(vw) = a, C(vz) = b \neq a, C(wx) = b, C(zx) = C(xy) = a, C(zy) = b, C(uw) = c, c \neq a, c \neq b$, there exists a (wy, b)-path, there exists a (wy, a)-path which does not go through the vertex x and there exists a (zu, a)-path which does not go through the vertex x. In this case we recolor $C(uw) = a, C(uv) = c, C(vw) = b, C(vz) = a, C(zx) = b, C(wx) = c$ as in Figure 18. We do not introduce any monochromatic cycle because colors a, b, c are pairwise different and all the a-edges of the configuration are on an a-path which ends at v and x.

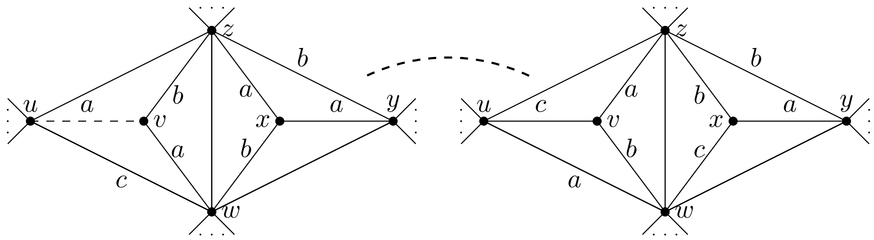


Figure 18: Case 3.1.2.2.3.1 in the proof of Lemma 12

Case 3.1.2.2.3.2 $C(vw) = a, C(vz) = b \neq a, C(wx) = b, C(zx) = C(xy) = a, C(zy) = b, C(uw) = b$, there exists a (uy, b)-path, there exists a (wy, a)-path which does not go through the vertex x and there exists a (zu, a)-path which does not go through the vertex x. Let $c = C(wz)$. Since the vertex w has already two incident b-edges we have $c \neq b$. Because of the (wy, a)-path we have $c \neq a$ thus we can recolor $C(uw) = a, C(uv) = b, C(vw) = c, C(vz) = a, C(zx) = c, C(zw) = b$ as in Figure 19. We do not introduce any monochromatic cycle because colors a, b, c are pairwise distinct and all the a-edges of the configuration are on an a-path which ends at v and x.

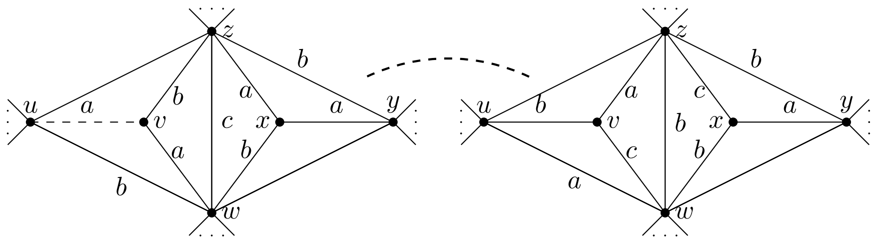


Figure 19: Case 3.1.2.2.3.2 in the proof of Lemma 12

Case 3.2 $C(vw) = a, C(vz) = b \neq a, C(wx) = a$ and there exists a (xu, a)-path, see Figure 20. Because of the (xu, a)-path the vertex x has exactly two incident edges colored a thus we consider which is the second one.

Case 3.2.1 $C(vw) = a, C(vz) = b \neq a, C(wx) = a, C(xy) = a$ and there exists a (uy, a)-path. Let $c = C(xz)$. Obviously $c \neq a$ since otherwise the vertex x would have three incident a-edges. Now we see what happens if $c \neq b$ and what if $c = b$.

Case 3.2.1.1 $C(vw) = a, C(vz) = b, C(wx) = a, C(xy) = a, C(xz) = c, a \neq b \neq c \neq a$ and there exists a (uy, a)-path. Note that we can assume that $C(wy) \neq c$, for otherwise we just

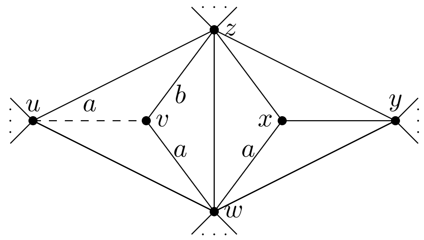


Figure 20: Case 3.2 in the proof of Lemma 12

swap the colors of zv and zx. Then we recolor edges wx and xy to $C(wy)$ and wy to a and we obtain the situation where $C(wx) \neq a$ which was already considered in Case 3.1.

Case 3.2.1.2 $C(vw) = a, C(vz) = b, C(wx) = a, C(xy) = a, C(xz) = b, a \neq b$ and there exists a (uy, a)-path. In this case we use the color $c = C(uw) \neq a$ to recolor $C(xw) = C(uv) = c$ and $C(uw) = a$ as in Figure 21. Since $a \neq b$ and $a \neq c$ we do not introduce any a-cycle. If $b \neq c$ we are done but if $b = c$ we can have a b-cycle if there exists a (uw, b)-path but it can not be true because it would imply that there is a b-cycle in G'.

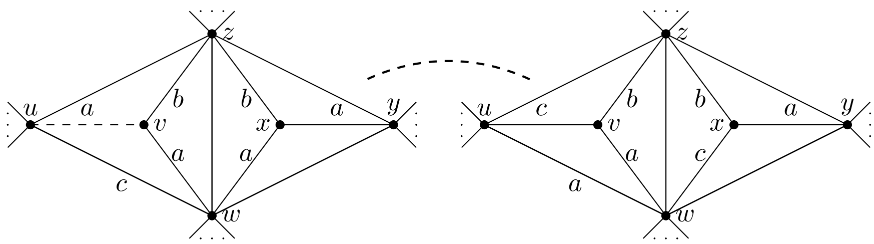


Figure 21: Case 3.2.1.2 in the proof of Lemma 12

Case 3.2.2 $C(vw) = a, C(vz) = b \neq a, C(wx) = a, C(xz) = a$ and there exists a (uz, a)-path. Let $c = C(wz)$. We may assume that $C(xy) = c$ since otherwise we can recolor $C(wx) = C(xz) = c$ and $C(wz) = a$ ending in a situation where $C(wx) \neq a$ which is considered in Case 3.1. We swap colors of edges zv and zx (as in Figure 22) and check whether we introduce a monochromatic cycle (without taking the edge uv into consideration).

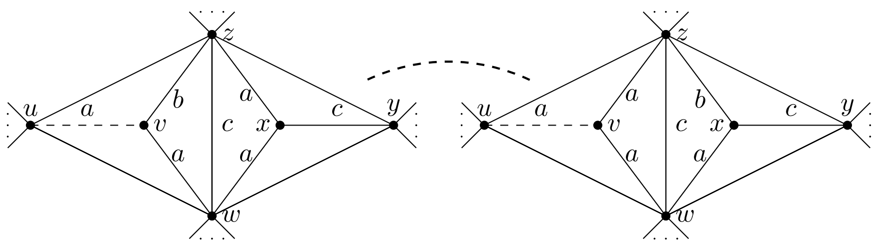


Figure 22: Case 3.2.2 in the proof of Lemma 12

Case 3.2.2.1 We do not introduce a monochromatic cycle thus we have a k-linear coloring satisfying $C(wv) = C(vz) = a$ which is considered in Case 2.

Case 3.2.2.2 We introduce a monochromatic cycle. The only possibility which leads to

Preprint series, IMFM, ISSN 2232-2094, no. 1111, January 19, 2010

a monochromatic cycle is when $b = c$ and there exists a (wy, b)-path. Now let us take into consideration edge uw colored $d = C(uw)$. We try to recolor $C(uv) = C(wx) = d$ and $C(uw) = a$ as in Figure 23. If there is no monochromatic cycle even when taking the edge uv into consideration we are done. Otherwise we can assume that $b = d$ and we consider subcases regarding the color of the edge uz.

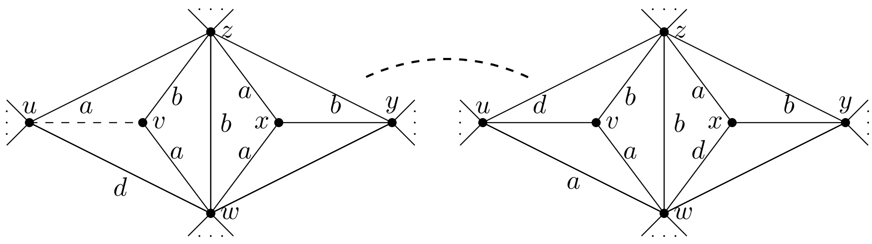


Figure 23: Case 3.2.2.2 in the proof of Lemma 12

Case 3.2.2.2.1 $C(vw) = C(wx) = C(xz) = a, C(xy) = C(vz) = C(zw) = C(wu) = b \neq a, C(uz) = a$. Let us consider the edge wy colored $c = C(wy)$. Since the vertex w already has two incident a-edges and two incident b-edges we have $c \neq a$ and $c \neq b$. In this case we recolor $C(uz) = b, C(uw) = a, C(vw) = c, C(wz) = a, C(wx) = b, C(wy) = b, C(xy) = c$ and put $C(uv) = a$ as in Figure 24 without introducing any monochromatic cycle.

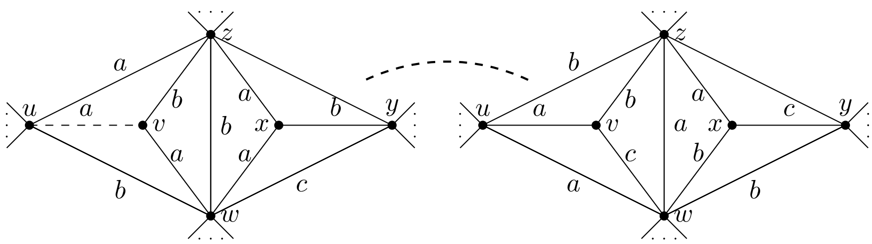


Figure 24: Case 3.2.2.2.1 in the proof of Lemma 12

Case 3.2.2.2.2 $C(vw) = C(wx) = C(xz) = a, C(xy) = C(vz) = C(zw) = C(wu) = b \neq a, C(uz) = c \neq a$. Since the vertex z already has two incident b edges we have $c \neq b$. In this case we recolor $C(uz) = b, C(zv) = a, C(zx) = c, C(uw) = a, C(vw) = b$ and put $C(uv) = c$ as in Figure 25. We do not introduce any monochromatic cycle since each of the recolored edges is on a monochromatic path which ends at v or x (see Fig. 25).

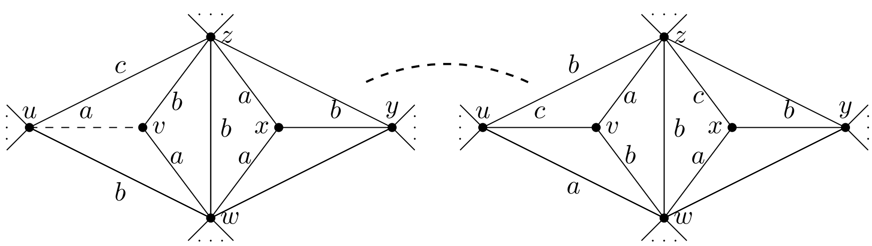


Figure 25: Case 3.2.2.2.2 in the proof of Lemma 12

Preprint series, IMFM, ISSN 2232-2094, no. 1111, January 19, 2010

□

Corollary 13. *G does not contain the configuration in Fig. 26.*

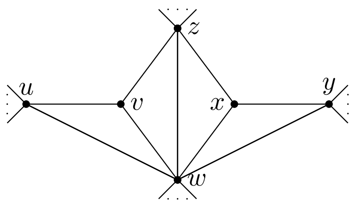


Figure 26: The configuration from Corollary 13. $\deg(v) = \deg(x) = 3$.

Proof. From Lemma 7 both $uz \in E(G)$ and $yz \in E(G)$. Hence we use Lemma 12. □

2.2 Proof of Proposition 3

Now we use the following lemma due to Cole, Kowalik and Škrekovski.

Lemma 14 (Proposition 1.3 in [6]). *Let G be a simple planar graph with minimum degree $\delta \geq 2$ such that each d-vertex, $d \geq 12$, has at most $d - 11$ neighbors of degree 2. Then G contains an edge of weight at most 13.*

Using the above lemma and Lemmas 4 and 6 only, we can prove the following special case of Proposition 3, which already improves known bounds on the linear arboricity of planar graphs.

Proposition 15. *Any simple planar graph of maximum degree Δ has a linear coloring in $\max\{\lceil\frac{\Delta}{2}\rceil, 6\}$ colors.*

Proof. Assume the claim is false and let G be a minimal counterexample (in terms of the number of edges). Let $k = \max\{\lceil\frac{\Delta}{2}\rceil, 6\}$. In particular, $\Delta \leq 2k$. By Lemma 4, G has no vertices of degree 1 and any 2-vertex has two neighbors of degree $2k$. Next, by Lemma 6 every $(2k)$-vertex has at most one 2-neighbor. Since $2k \geq 12$ the assumptions of Lemma 14 are satisfied, so G contains an edge of weight at most $13 \leq 2k + 1$, a contradiction with Lemma 4. □

Now, we proceed to the proof of Proposition 3. By the above proposition, Proposition 3 holds for $\Delta \geq 11$. Hence, in what follows we assume that $\Delta \leq 10$. We put $k = \max\{\lceil\frac{\Delta}{2}\rceil, 5\} = 5$ and assume that G is a minimal counterexample (in terms of the number of edges).

We prove Proposition 3 using the discharging method. The procedure is the following. We assign a number (called *charge*) to every vertex and face of a plane embedding of G, such that the total sum of all charges is negative. Next, we redistribute the charge among vertices and faces in such a way, that using the structural properties of the graph G described in Section 2.1 we are able to show that every vertex and every face has a nonnegative charge at the end, hence the total charge of G is nonnegative. This will give a contradiction, so the minimal counterexample does not exist.

Initial charge. We set the following initial charge to all vertices and faces of G:

$$\begin{aligned} \mathrm{ch}_0(v) &= \deg(v) - 4\,, \;\; v \in V(G)\,, \\ \mathrm{ch}_0(f) &= \ell(f) - 4\,, \;\; f \in F(G)\,. \end{aligned}$$

From the Euler's formula we infer the following total charge of G:

$$\begin{aligned} \sum_{v \in V(G)} (\deg(v) - 4) + \sum_{f \in F(G)} (\ell(f) - 4) = \\ 2|E(G)| - 4|V(G)| + 2|E(G)| - 4|F(G)| = \\ -4(|V(G)| - |E(G)| + |F(G)|) = -8. \end{aligned}$$

Discharging rules. Now, we present discharging rules, by which we redistribute the charge of vertices and faces in G.

(R1) Every 10-vertex sends 1 to an adjacent 2-vertex.

(R2) Every (≥ 9)-vertex sends $\frac{1}{3}$ to every adjacent 3-vertex.

(R3) Every (≥ 8)-vertex sends $\frac{1}{2}$ to every incident 3-face with a vertex of degree at most 4.

(R4) Every (≥ 7)-vertex sends $\frac{2}{5}$ to every incident 3-face with a 5-vertex.

(R5) Every (≥ 6)-vertex sends $\frac{1}{3}$ to every incident 3-face which is incident to only (≥ 6)-vertices.

(R6) Every 5-vertex sends $\frac{1}{5}$ to every incident 3-face.

(R7) Every ($\geq$5)-face f sends $\frac{1}{3}$ to every incident 10-vertex which has a 2-neighbor incident to f.

Final charge. Note that the initial charge is negative only for 2- and 3-vertices, and for 3-faces. We show that by applying the discharging rules, all vertices and faces of G have nonnegative final charge.

First, we consider the charge of faces. Note that 4-faces do not send any charge so their charge remains 0. Now we consider a face f of length $\ell(f) \geq 5$. By Lemmas 4 and 6, f is incident to at most $\lfloor \frac{\ell(f)}{3} \rfloor$ vertices of degree 2. Hence f sends at most $\frac{1}{3} \cdot 2 \cdot \lfloor \frac{\ell(f)}{3} \rfloor$ units of charge by (R7), which is less than $\ell(f) - 4$ for $\ell(f) \geq 5$, hence f retains positive charge.

It only remains to show that every 3-face f receives at least 1 from its neighbors, since its initial charge is -1. We consider cases regarding the degree of vertices incident to f. If f is incident to a 2-, 3-, or 4-vertex, it follows by Lemma 4 that the other two vertices incident to f are of degree at least $2k - 2 \geq 8$, hence each of them sends $\frac{1}{2}$ to f by (R3). Next, if f is incident to a 5-vertex v, the other two incident vertices of f are of degree at least $2k - 3 \geq 7$ by Lemma 4. Hence, f receives $\frac{1}{5}$ from v by (R6) and $\frac{2}{5}$ from each of the other two incident vertices by (R4), that is 1 in total. Finally, if f is incident only to $\geq$6-vertices, each of them sends $\frac{1}{3}$ by (R5), hence f receives 1 in total again. It follows that the final charge of 3-faces is 0.

Now, we consider the final charge of vertices. For convenience, we introduce a notion of a side. Let v be a vertex and let $vx_0, \ldots, vx_{\deg(v)-1}$ be the edges incident to v, enumerated

Preprint series, IMFM, ISSN 2232-2094, no. 1111, January 19, 2010

in the clockwise order around v in the given plane embedding. For any $i = 0, \ldots, \deg(v) - 1$, the pair $s = (vx_i, vx_{i+1})$ will be called a *side of* v (where $x_{\deg(v)} = x_0$). If x_i and x_{i+1} are adjacent, we say that s is *triangular*. We also say that s is incident to vx_i and vx_{i+1}. Note that v can have less than $\deg(v)$ incident faces (when v is a cutvertex), while it has always $\deg(v)$ distinct incident sides. However, for each triangular face incident to v there is a distinct triangular side of f. Since v does not send charge to non-triangular faces, when v sends charge to a triangle we can say that it sends the charge to the corresponding side and the total charge sent to sides is equal to the total charge sent to faces. In what follows, we use the following claim.

Claim 1 *If a d-vertex v has a negative final charge and v is not adjacent to a 2-vertex then v has at most $11 - d$ non-triangular sides.*

Proof (of the claim). Let p be the number of non-triangular sides of v. Note that v sends charge only to incident triangles (at most $\frac{1}{2}$ per triangle) and to adjacent 3-vertices (at most $\frac{1}{3}$ per 3-vertex). For the proof of this claim, we replace (R2) by an equivalent rule:

(R2') For each 3-neighbor w of a (≥ 9)-vertex v, vertex v sends $1/6$ to each of the two sides incident to edge vw and each of these sides resend the $1/6$ to w.

Then, v sends at most $\frac{2}{3}$ to each incident triangular side (the corresponding 3-face has only one 3-vertex, for otherwise there is an edge of weight $6 < 2k + 2$, which contradicts Lemma 4) and it sends at most $\frac{1}{3}$ to each incident non-triangular side. It follows that v sends at most $\frac{1}{3} \cdot p + \frac{2}{3} \cdot (d - p) = \frac{2d-p}{3}$ in total. Hence, the final charge at v is negative when $\frac{2d-p}{3} > d - 4$, which is equivalent to $p < 12 - d$ and Claim 1 follows since p is a natural number. □

Now we consider several cases regarding the degree of vertex v.

- *v is a 2-vertex.* The initial charge of v is -2. By Lemma 4, both its neighbors are of degree at least $2k \geq 10$. Hence, by (R1), v receives 1 from each of the two neighbors, and since it does not send any charge, its final charge is 0.

- *v is a 3-vertex.* The initial charge of v is -1. By Lemma 4, all three of its neighbors are of degree at least $2k - 1 \geq 9$. By (R2), v receives $\frac{1}{3}$ from each of the three neighbors, and since it does not send any charge, its final charge is 0.

- *v is a 4-vertex.* In this case, v does not send nor receive any charge. Hence, its initial charge, which is 0, is equal to its final charge.

- *v is a d-vertex,* $5 \leq d \leq 8$. Note that v sends charge only to incident triangles. By rules (R3)-(R6), the charge v sends to each incident triangle is at most $\frac{1}{5}$, $\frac{1}{3}$, $\frac{2}{5}$, $\frac{1}{2}$, for $d = 5, 6, 7, 8$ respectively. One can check that in each of the four cases this is not more than $\frac{d-4}{d}$, and since there are at most d incident triangles, v sends at most $d - 4$ charge in total and retains nonnegative charge.

- *v is a 9-vertex.* The initial charge at v is 5. If v has at most one 3-neighbor then v sends at most $\frac{9}{2}$ to faces and $\frac{1}{3}$ to vertices so its final charge is positive. If v has at least two 3-neighbors, then by Lemma 11, each of them is incident to two non-triangular sides of v. Hence v has at least 3 non-triangular sides, which contradicts Claim 1.

Preprint series, IMFM, ISSN 2232-2094, no. 1111, January 19, 2010

- *v is a 10-vertex.* Assume first that v has no 2-neighbors. By Claim 1, v is incident to at most one non-triangular side. If v is incident only to 3-faces, by Corollary 13 v has at most three 3-neighbors, since otherwise a pair of 3-neighbors would have two common neighbors. Then, v sends at most $3 \cdot \frac{1}{3} = 1$ to vertices and at most $10 \cdot \frac{1}{2} = 5$ to faces, hence at most 6 in total. If v is incident to one non-triangular side, by Corollary 13 v has at most four 3-neighbors. Then, v sends at most $4 \cdot \frac{1}{3} = \frac{4}{3}$ to vertices and at most $9 \cdot \frac{1}{2} = \frac{9}{2}$ to faces, that is less than 6 in total. Hence, in both cases the final charge at v is nonnegative.

  Finally, assume that v has a 2-neighbor. Let $w_0, w_1, \ldots, w_9$ denote the neighbors of v in the clockwise order in the given plane embedding of G. Assume w.l.o.g. $\deg(w_1) = 2$. By Lemma 6, this is the only 2-neighbor of v. By Lemma 5, the neighbors of w_1 are adjacent, so assume w.l.o.g. w_0 is adjacent to w_1 (in the beginning, we can choose the plane embedding of G so that each triangle with one 2-vertex and two 10-vertices is a face). Since G is simple, the face incident to vw_1 and vw_2, say f, is of length at least 4.

  Let n_3 denote the number of 3-vertices among vertices $w_3, \ldots, w_9$. By Corollary 10, each of these 3-neighbors is incident to at least one non-triangular side. Since each side is incident to at most two 3-neighbors of v, there are at least $\lceil \frac{n_3}{2} \rceil$ non-triangular sides, not counting the side (vw_1, vw_2).

  It follows that v sends 1 unit to w_1, $\frac{1}{3}$ to w_2 if $\deg(w_2) = 3$, $\frac{n_3}{3}$ to vertices $w_3, \ldots, w_9$ and at most $\frac{1}{2} \cdot (9 - \lceil \frac{n_3}{2} \rceil)$ to incident faces. Hence, v sends at most $5\frac{1}{2} + \frac{1}{3}[\deg(w_2) = 3] + \frac{n_3}{3} - \frac{1}{2}\lceil \frac{n_3}{2} \rceil$. However, when $\deg(w_2) = 3$, then face f is of length at least 5 by Lemma 8, so v receives additional $\frac{1}{3}$ from f by (R7). Hence, v gets at least $6 + \frac{1}{3}[\deg(w_2) = 3]$ charge in total. It follows that the final charge at v is at least $\frac{1}{2} - \frac{n_3}{3} + \frac{1}{2}\lceil \frac{n_3}{2} \rceil$, which is nonnegative since $n_3 \leq 7$.

It follows that the total charge of G is nonnegative, establishing a contradiction with the existence of a minimal counterexample.

3 Algorithm

In this section we show that our proof can be turned into an efficient algorithm for finding a linear $\lceil \frac{\Delta}{2} \rceil$-colorings. The forbidden subgraphs from Section 2.1 will be called *reducible configurations*. It should be clear that the proof of Proposition 3 corresponds to the following algorithm: find any of our reducible configurations in linear time, then obtain a smaller graph in constant time by removing/contracting an edge, color it recursively and finally extend the coloring in linear time. Since in each recursive call the number of edges decreases, the number of recursive calls is linear, which gives $O(n^2)$ overall time complexity. However, with some effort it is possible to improve the running time. Namely, we present an $O(n \log n)$-time algorithm. The algorithm works for any planar graph and returns a partition into $\max\{\lceil \frac{\Delta}{2} \rceil, 5\}$ linear forests, which is optimal for $\Delta \geq 9$.

Our approach is as follows. First we describe an algorithm that finds a partition into $\max\{\lceil \frac{\Delta}{2} \rceil, 6\}$ linear forests, which is optimal for $\Delta \geq 11$. This can be treated as an implementation of Proposition 15. Recall that for proving this proposition we needed only a few reducible configurations: an edge of weight at most $2k + 1$, a 2-vertex with its neighbors nonadjacent, and a $2k$-vertex with two 2-neighbors. As we will see in Subsection 3.1 these configurations are simple enough to find them very fast (even in constant time) after a linear preprocessing. Once we have this algorithm, we use it whenever $\Delta \geq 11$. Otherwise $\Delta \leq 10$, so Δ is bounded which makes finding any bounded-size configuration very easy. Then we use the algorithm sketched in Subsection 3.2.

3.1 An algorithm for $\Delta \geq 11$

The coloring algorithm we describe in this section is inspired by the linear-time algorithm for the Δ-edge-coloring of planar graphs presented in [5]. For an input graph G of maximum degree Δ we define $k = \max\{\lceil \frac{\Delta}{2} \rceil, 6\}$. We will show an $O(n \log n)$-time algorithm which finds a k-linear coloring of G.

Note that $\Delta \leq 2k$, so the graph has vertices of degree at most $2k$ and $k \geq 6$ (we will use these facts in our arguments). We use the following three types of reducible edges of weight at most $2k+1$, which will be called *nice*:

- edges of weight at most 13,
- edges incident to a 1-vertex, and
- edges incident to a 2-vertex and a vertex of degree at most $2k-1$.

Our algorithm uses two queues: Q_e and Q_2. The queue Q_e stores nice edges, while the queue Q_2 stores 2-vertices such that their both neighbors are, or used to be, of degree $2k$. Also, any $(2k)$-vertex x may store a triangle xyz, such that $\deg(y) = 2$ and $\deg(z) = 2k$. During the execution of the algorithm the following invariants are satisfied.

Invariant 1. *For every nice edge uv, either uv stored in Q_e, or one of its endpoints is of degree 2 and is stored in Q_2.*

Invariant 2. *For each 2-vertex x with two $(2k)$-neighbors v and w, either x is Q_2 or G contains a triangle vxw and this triangle is stored in both v and w. Each vertex stores at most one triangle.*

It is easy to initialize the queues in linear time to make invariants 1 and 2 satisfied at the beginning (without storing triangles in vertices). Then we use a recursive procedure which can be sketched as follows. By configuration A and B we mean the configurations from the cases A and B of the proof of Lemma 6 (see Fig. 1).

Step 1. (Base of the recursion.) If G has no edges, return the empty coloring.

Step 2. If Q_e contains an edge e, obtain a coloring of $G-e$ recursively and color e by a free color as described in Lemma 4. Return the resulting coloring of G.

Step 3. Remove a vertex x from Q_2 until $\deg(x) = 2$. Denote the neighbors of x by v and w.

Step 4. If $\deg(v) < 2k$ (resp. $\deg(w) < 2k$), add xv (resp. xw) to Q_e and go to Step 2.

Step 5. If v or w stores a triangle vwy, we have a configuration A. Remove an edge e of G as described in Lemma 6, recurse on $G-e$ and extend the coloring of $G-e$ to a coloring of G as described in Lemma 6.

Step 6. If $vw \notin E(G)$, proceed as in Lemma 5: remove the vertex x and add an edge vw, recurse, add vertex x and edges vx, wx, color these edges the same color as vw and remove the edge vw.

Step 7. Else ($vw \in E(G)$)

(i) If v (resp. w) stores a triangle vyu, we have configuration B. Remove an edge e of G as described in Lemma 6, recurse on $G - e$ and extend the coloring of $G - e$ to a coloring of G as described in Lemma 6.

(ii) Otherwise, store the triangle vxw in v and w.

Now we describe how the queues Q_e and Q_2 are updated during an execution of the algorithm, to keep invariants 1 and 2 satisfied. First notice that it is easy to store degrees of vertices and update them in overall $O(n)$ time. Then, whenever an edge is removed, for each of its endpoints, say z, we check whether z is of degree at most 12. If so, for all its $O(1)$ incident edges we check whether they are nice and if that is the case we add them to Q_e (unless Q_e already contains this edge). Also, if the degree of z was decreased from $2k$ to $2k-1$, and z stored a triangle zxy with $\deg(x) = 2$ and $\deg(y) = 2k$, then we add the edge zx to Q_e and remove the triangle from both z and y. Also, when after removing an edge a degree of its endpoint z drops to 1, we check whether z is on Q_2 and if so, we remove z from Q_2. Now it is easy to see that Invariant 1 is satisfied. Finally, when after removing an edge a degree of its endpoint z drops to 2, we check whether both of its neighbors are of degree $2k$ and if so, we add z to Q_2. This makes Invariant 2 satisfied as well. Notice that the updates described above take only $O(1)$ time after each edge deletion. Clearly, after the graph modification in Step 6, there is no need to update any queue.

Now we are going to show the correctness of our algorithm.

Proposition 16. *Let $k = \max\{\lceil\frac{\Delta}{2}\rceil, 6\}$. The above algorithm correctly finds a k-linear coloring of any planar graph of maximum degree Δ.*

Proof. Clearly, it suffices to show that whenever the algorithm finds itself in Step 3 the queue Q_2 is not empty. Assume the contrary. Since Q_e and Q_2 are empty, by Invariant 1 there are no nice edges, so G has no 1-vertices and each 2-vertex is adjacent to two $(2k)$-vertices. Hence only $(2k)$-vertices have 2-neighbors. Since Q_2 is empty, by Invariant 2 each $(2k)$-vertex has at most one 2-neighbor. Hence the assumptions of Lemma 14 are satisfied and G contains an edge of weight at most 13. But this edge is nice and we get the contradiction with Invariant 1 and the fact that Q_e is empty. □

Proposition 17. *The above algorithm can be implemented in $O(n \log n)$ time.*

Proof. First we show that each recursive call takes only $O(\log n)$ amortized time. Checking adjacency in Step 6 can be easily done in $O(\log n)$ time e.g. by storing the neighbors of each vertex in a balanced tree. Then adding and removing edges can be done in $O(\log n)$ time. It remains to consider recoloring the graph after going back from the recursion. Recall from the proof of Lemma 6 that during the recoloring the algorithm checks colors of a bounded number of edges and also recolors a bounded number of edges. Finding a free color can be done in $O(\log n)$ time using any balanced binary search tree containing used colors. When we are looking for a free color we can go down from the root of the tree going into the subtree which does not contain all colors from its range. To achieve it in each node of the balanced tree we store the size of its subtree.

The last unclear issue is verifying whether there is a path of given color, say a, between two vertices, say x, y. W.l.o.g. we can assume that both x and y are incident to an edge colored a (otherwise, immediately, the answer in negative), so in fact, given two edges of the same color we want to check whether they are on the same path in the linear forest of color a. Note that during recoloring an edge of color a to b (say), some path in the linear forest of a is split into two paths (possibly one of length 0), and some two paths (possibly empty) of the linear forest of color b are connected to one path. In other words, we need a data structure which maintains a linear forest that can be updated after adding or removing an edge and processes connectivity queries of the form "are the edges e_1 and e_2 on the same path?". There are several solutions to this problem generalized to forests with $O(\log n)$ time complexity (amortized) both for updates and queries – e.g. link-cut trees of Sleator and Tarjan [12] or ET-trees of Henzinger and King [10]. We note that in the case of *linear* forests this time complexity can be also achieved by using simply a balanced BST tree with efficient merge and split operations, like e.g. splay trees. All these data structures take only linear space with respect to the size of the linear forest.

Since in a one recursive call we perform a bounded number of path queries, this takes only $O(\log n)$ amortized time. □

3.2 An algorithm for $\Delta \leq 10$

Now we sketch an algorithm which finds a partition of any planar graph of maximum degree $\Delta = O(1)$ into $k = \max\{\lceil \frac{\Delta}{2} \rceil, 5\}$ linear forests. Our algorithm uses all the reducible configurations described in Section 2.1. Recall that they are of bounded size. Hence it is easy to check in constant time, whether a given vertex v belongs to a given configuration, since if this is the case, this configuration is a subgraph of the graph induced by all vertices at some bounded distance from v and because $\Delta = O(1)$ this subgraph has a bounded size. Our algorithm uses a queue of reducible configurations, initialized in linear time. Then configurations are added to the queue after modifying the graph. Since each modification decreases the size of the graph, and causes appearance of a bounded number of configurations, the total number of configurations is linear. After finding a configuration (by just removing it from the queue in constant time), shrinking the graph (usually by removing an edge), and going back from the recursive call, extending the coloring of the shrinked graph to the original graph takes $O(\log n)$ time, as described in the proof of Proposition 17. It may happen that when we remove a configuration from the queue it is no longer a configuration because we have erased some parts of if in the meantime. In such a situation we simply skip this configuration and take another one from the queue.

Corollary 18. *Let $k = \max\{\lceil \frac{\Delta}{2} \rceil, 5\}$. The above algorithm finds a k-linear coloring of any planar graph of maximum degree Δ in $O(n \log n)$ time.*

References

[1] J. Akiyama, G. Exoo, and F. Harary. Covering and packing in graphs III: Cyclic and acyclic invariants. *Math Slovaca*, 30:405–417, 1980.

[2] J. Akiyama, G. Exoo, and F. Harary. Covering and packing in graphs IV: Linear arboricity. *Networks*, 11:69–72, 1981.

[3] N. Alon. The linear arboricity of graphs. *Israel Journal of Mathematics*, 62(3):311–325, 1988.

[4] N. Alon, V. Teague, and N. C. Wormald. Linear arboricity and linear k-arboricity of regular graphs. *Graphs and Combinatorics*, 17(1):11–16, 2001.

[5] R. Cole and L. Kowalik. New linear-time algorithms for edge-coloring planar graphs. *Algorithmica*, 50(3):351–368, 2008.

[6] R. Cole, L. Kowalik, and R. Škrekovski. A generalization of Kotzig's theorem and its application. *SIAM Journal on Discrete Mathematics*, 21(1):93–106, 2007.

[7] F. Guldan. The linear arboricity of 10-regular graphs. *Math Slovaca*, 36(3):225–228, 1986.

[8] B. P. H. Enomoto. The linear arboricity of some regular graphs. *J. Graph Theory*, 8:309–324, 1984.

[9] F. Harary. Covering and packing in graphs I. *Ann. N.Y. Acad. Sci*, 175:198–205, 1970.

[10] M. R. Henzinger and V. King. Randomized fully dynamic graph algorithms with polylogarithmic time per operation. *J. ACM*, 46(4):502–516, 1999.

[11] B. Peroche. Complexity of the linear arboricity of a graph. *RAIRO Oper. Res.*, 16:125–129, 1982. In French.

[12] D. D. Sleator and R. E. Tarjan. A data structure for dynamic trees. *J. Comput. Syst. Sci.*, 26(3):362–391, 1983.

[13] V. G. Vizing. Critical graphs with a given chromatic number. *Diskret. Analiz*, 5:9–17, 1965.

[14] J. L. Wu. On the linear arboricity of planar graphs. *J. Graph Theory*, 31:129–134, 1999.

[15] J.-L. Wu, J.-F. Hou, and G.-Z. Liu. The linear arboricity of planar graphs with no short cycles. *Theor. Comput. Sci.*, 381(1-3):230–233, 2007.

[16] J.-L. Wu, J.-F. Hou, and X.-Y. Sun. A note on the linear arboricity of planar graphs without 4-cycles. In *International Symposium on Operations Research and Its Applications*, pages 174–178, 2009.

[17] J. L. Wu and Y. W. Wu. The linear arboricity of planar graphs of maximum degree seven is four. *J. Graph Theory*, 58(3):210–220, 2008.