Discrete Mathematics 341 (2018) 820–824

Contents lists available at ScienceDirect

Discrete Mathematics

journal homepage: www.elsevier.com/locate/disc

Note

Multi-set neighbor distinguishing 3-edge coloring

Bojan Vučković

Mathematical Institute, Serbian Academy of Science and Arts, Kneza Mihaila 36 (P.O. Box 367), 11001 Belgrade, Serbia

ARTICLE INFO

Article history:
Received 18 October 2016
Received in revised form 7 July 2017
Accepted 1 December 2017

Keywords:
Multi-set neighbor distinguishing edge coloring

ABSTRACT

Let G be a graph without isolated edges, and let $c : E(G) \to \{1, \dots, k\}$ be a coloring of the edges, where adjacent edges may be colored the same. The color code of a vertex v is the ordered k-tuple $(a_1, a_2, \dots, a_k)$, where a_i is the number of edges incident with v that are colored i. If every two adjacent vertices of G have different color codes, such a coloring is called multi-set neighbor distinguishing. In this paper, we prove that three colors are sufficient to produce a multi-set neighbor distinguishing edge coloring for every graph without isolated edges.

1. Introduction

All graphs considered in this paper are finite and simple. Let $V(G)$ and $E(G)$ denote the vertex set and the edge set of a graph G, respectively. Let $N(v)$ denote the set of neighbors of a vertex v and $d(v)$ denote the degree of v. A dominating set S of a graph G is a set of vertices such that each vertex of $V(G)$ is either in the set S or is adjacent in G to a vertex of S. A set of vertices is independent if no two vertices in the set are adjacent. For a subset S of $V(G)$, the subgraph $G[S]$ of G induced by S has S as its vertex set and two vertices of S are adjacent in $G[S]$ if and only if they are adjacent in G. For notation not defined here, we refer the reader to [2].

A graph G is *normal* if it contains no isolated edges. Let G be a normal graph, and let $c : E(G) \to \{1, \dots, k\}$ be a coloring of the edges, where adjacent edges may be colored the same. If c uses k colors we say that c is a *k-edge coloring*. The *color code* of a vertex v is the ordered k-tuple $\mathrm{code}(v) = (a_1, \dots, a_k)$, where a_i is the number of edges incident with v that are colored i. Thus $\sum_{i=1}^{k} a_i = d(v)$. If every two adjacent vertices of a graph G have different color codes, such a coloring is called *multi-set neighbor distinguishing*. Since a graph that contains isolated edges does not accept such a coloring, we only consider normal graphs.

Karoński, Łuczak, and Thomason proved in [3] that if a normal graph G has chromatic number at most 3, it is possible to color the edges with the colors 1, 2 and 3, so that for every two adjacent vertices u and v of G, the sum of the colors of the edges incident with u is different from the sum of the colors of the edges incident with v. This implies that for an arbitrary normal graph G with chromatic number at most 3, three colors are sufficient to produce a multi-set neighbor distinguishing edge coloring. The best result concerning an arbitrary normal graph stating that four colors are sufficient to produce such a coloring was proved by Addario-Berry et al. in [1]. We improve this result in Theorem 1 by showing that, in fact, three colors are sufficient.

Theorem 1. *For every normal graph G, there exists a multi-set neighbor distinguishing 3-edge coloring of G.*

Note that, in general case, three colors are necessary. There are many graphs that do not allow a multi-set neighbor distinguishing edge coloring with less than three colors. Examples of such graphs are K_n when $n \geq 3$, and C_n when $n \geq 3$ and $n \not\equiv 0 \bmod 4$.

E-mail address: b.vuckovic@turing.mi.sanu.ac.rs.

https://doi.org/10.1016/j.disc.2017.12.001

If the colors of some of the edges incident with v are left unassigned, a color code that includes the number of already colored edges incident with v is called a *partial color code*, and we denote it by $\mathrm{code}_p(v)$. In order to prove Theorem 1, we need the following lemma.

Lemma 2. *Let G be an induced subgraph of a graph H, where G is a connected bipartite graph with partite sets V_1 and V_2. Let there be given a partial edge coloring ϕ of H with arbitrary colors assigned to every edge uv with $u \in V(G)$ and $v \in V(H) \setminus V(G)$. Then, for arbitrary $v \in V_1$, there exists an extension of the coloring ϕ such that all the edges of G are assigned colors from $\{a, b\}$ and all the vertices of $V_1 \setminus \{v\}$ have in H an even (odd) number of incident edges colored a, while all the vertices of V_2 have in H an odd (even, respectively) number of incident edges colored a.*

Proof. First, we color all the edges of G with the color a. While there exist two or more vertices in $V_1 \setminus \{v\}$ that are incident with an odd number of edges colored a, we do the following. Let w_1 and w_2 be two such vertices. We interchange the colors a and b for every edge on a path between w_1 and w_2 in G. This way the parity of the number of edges incident with w_1 and w_2 colored a is changed, while the parity of the number of edges colored a incident with any other vertex of G stays the same. At the end of this procedure, there is at most one vertex in $V_1 \setminus \{v\}$ that has an odd number of incident edges colored a. Next, while there exist two or more vertices in V_2 that are incident with an even number of edges colored a, we do the following. Let u_1 and u_2 be two such vertices. We interchange the colors a and b for every edge on a path between u_1 and u_2 in G. At the end of this procedure there is at most one vertex in V_2 that has an even number of incident edges colored a. If there are vertices $w \in V_1 \setminus \{v\}$ with an odd number of incident edges colored a, and $u \in V_2$ with an even number of incident edges colored a, then we interchange the colors a and b for every edge on a path between w and u in G. Now, there might be only one vertex y of $(V_1 \setminus \{v\}) \cup V_2$ with the undesired parity of the number of incident edges colored a. If such a vertex exists we interchange the colors a and b for every edge on a path between y and v in G, thus obtaining the desired edge coloring. The proof is analogous for the case when we want to obtain that all the vertices of $V_1 \setminus \{v\}$ have an odd number, and the vertices of V_2 have an even number of incident edges colored a. □

2. Proof of Theorem 1

Proof of Theorem 1. If the statement of Theorem 1 holds for any connected graph with at least two edges, than it clearly holds for any normal graph. Thus we may assume that G is a connected graph with two or more edges. Let $V = V(G) = V_1 \cup \cdots \cup V_k$, where V_1 is an independent dominating set of G, and V_i is an independent dominating set of

$$G[V \setminus (\bigcup_{j<i} V_j)]$$

for every $1 < i < k$, while V_k is an independent set. Thus for every $v \in V_i$ with $1 < i \le k$, the vertex v has at least one neighbor in every set V_j with $1 \le j < i$, and none in the set V_i. Let v be a vertex of V_i with $\mathrm{code}(v) = (b_1, b_2, b_3)$. The idea of the proof is to color the edges of G in such a way that, except in a few special cases, the following properties hold:

1. if $i = 2l + 3$ with $l \ge 1$, then $b_1 \ge 1$, $b_2 = l$ and b_3 is an odd number greater than 1,
2. if $i = 2l + 2$ with $l \ge 1$, then $b_1 = l$, $b_2 \ge 1$ and b_3 is an even number greater than 0,
3. if $i = 3$, then $b_1 = 0$, $b_2 \ge 1$ and $b_3 \ge 1$,
4. if $i = 2$ and v has a neighbor in V_j with $j > 2$, then $b_1 + b_2 \ge 2$ and $b_3 = 0$,
5. if $i = 1$ and v has a neighbor in V_j with $j > 2$, then $b_1 \ge 0$, $b_2 = 0$ and $b_3 \ge 1$,
6. if $v \in V_1 \cup V_2$ and v has no neighbor outside of $V_1 \cup V_2$, we only take care that v has a different color code from all of its neighbors.

First, we assign the color 3 to all the edges joining the vertices of V_1 with the vertices of V_i, for every integer i with $3 \le i \le k$. Next, we color all the edges incident with the vertices of V_k, proceed with coloring of the edges incident with the vertices of V_{k-1}, and gradually decrease to V_2.

Let us assume that we have colored all the edges incident with the vertices of V_l, for all l with $i < l \le k$. We now assign colors to the edges joining the vertices of V_i and V_j, for all j with $1 \le j < i$. For each $v \in V_i$ we proceed as follows. Let $\mathrm{code}_p(v) = (a_1, a_2, a_3)$. We color the edges incident with v, depending on the value of i.

1. **For** $i > 5$:
   We consider two cases, depending on the parity of i.
   (a) $i = 2l + 3$, for some $l \ge 2$. Since v has at least one neighbor in V_1, we have $a_3 \ge 1$. For $i = k$ we have $a_1 = a_2 = 0$, and for $i < k$ we have $a_1 \ge 0$ and $a_2 = 0$. Colors have already been assigned to all the edges joining v with the vertices of V_1 and V_j with $j > i$. We now assign colors to the edges joining v with the vertices of V_j, for every j with $4 < j < i$:
      i. when j is odd, we color all such edges with 1,
      ii. when j is even, we color an edge between v and one of the vertices of V_j (such a vertex always exists) with 2, and all the remaining edges with 3.