Algorithmic Complexity of Proper Labeling Problems

Ali Dehghan\textsuperscript{a}, Mohammad-Reza Sadeghi\textsuperscript{a}, Arash Ahadi\textsuperscript{b}

\textsuperscript{a}Faculty of Mathematics and Computer Science, Amirkabir University of Technology, Tehran, Iran
\textsuperscript{b}Department of Mathematical Sciences, Sharif University of Technology, Tehran, Iran

Abstract

A proper labeling of a graph is an assignment of integers to some elements of a graph, which may be the vertices, the edges, or both of them, such that subject to some conditions we obtain a proper vertex coloring via the labeling. The problem of proper labeling offers many variants and received a great interest during these last years. In this work, we consider the computational complexity of some variants of the proper labeling problems such as: multiplicative vertex-coloring, fictional coloring and gap coloring. For instance, we show that, for a given bipartite graph $G$, determining whether $G$ has a vertex-labeling by gap from $\{1,2\}$ is \textbf{NP}-complete. Also, we prove that there is a polynomial time algorithm for determining whether a given planar bipartite graph $G$ has a vertex-labeling by gap from $\{1,2\}$. In sharp contrast, it is \textbf{NP}-complete to decide whether a given planar 3-colorable graph $G$ has a vertex-labeling by gap from $\{1,2\}$.

Key words: Proper Labeling; Multiplicative vertex-coloring weightings; Gap vertex-distinguishing edge colorings; Fictional Coloring; Computational Complexity.

Subject classification: 05C15, 05C20, 68Q25

1 Introduction

A proper labeling of a graph is an assignment of integers to some elements of a graph, which may be the vertices, the edges, or both of them, such that subject to some conditions we obtain a proper vertex coloring via the labeling. Karoński, Luczak and Thomason initiated the study of proper-labelings \cite{16}. They introduced an edge-labeling which is additive vertex-coloring that means for every edge $uv$, the sum of labels of the edges incident to $u$...
is different than the sum of labels of the edges incident to \( v \) [16]. The problem of proper labeling offers many variants and received a great interest during these last years, for instance see [1, 7, 8, 15, 16, 20]. First, consider the following two famous variants.

**(P1) Edge-labeling by sum.**

An edge-labeling \( f \) is edge-labeling by sum if \( c(v) = \sum_{e \ni v} f(e) \), \( \forall v \in V \) is a proper vertex coloring. This parameter was introduced by Karoński et al. and it is conjectured that three integer labels \{1, 2, 3\} are sufficient for every connected graph, except \( K_2 \) (1, 2, 3-Conjecture, see [16]). This labeling have been studied extensively by several authors, for instance see [1, 2, 6, 17, 20]. Currently, we know that every connected graph has an edge-labeling by sum, using the labels from \{1, 2, 3, 4, 5\} [15]. Also, it is shown that determining whether a given graph has a edge-labeling by sum from \{1, 2\} is NP-complete [12].

**(P2) Vertex-labeling by sum** (Lucky labeling and sigma coloring).

A vertex-labeling \( f \) is vertex-labeling by sum if \( c(v) = \sum_{u \sim v} f(u) \), \( \forall v \in V \) is a proper vertex coloring. vertex-labeling by sum is a vertex versions of the above problem, which was introduced recently by Czerwiński et al. [8]. It was conjectured that every graph \( G \) has a vertex-labeling by sum, using the labels \{1, 2, \cdots, \chi(G)\} [8] and it was shown that every graph \( G \) with \( \Delta(G) \geq 2 \), has a vertex-labeling by sum, using the labels \{1, 2, \cdots, \Delta^2-\Delta+1\} [4], also, it was shown that, it is NP-complete to decide for a given planar 3-colorable graph \( G \), whether \( G \) has a vertex-labeling by sum from \{1, 2\} [3]. Furthermore, it is NP-complete to determine for a given 3-regular graph \( G \), whether \( G \) has a vertex-labeling by sum from \{1, 2\} [10]. A similar version of this labeling was introduced by Chartrand et al. [7].

In this work, we consider the algorithmic complexity of the following proper labeling problems.

**(P3) Edge-labeling by product.** (Multiplicative vertex-coloring)

An edge-labeling \( f \) is edge-labeling by product if \( c(v) = \prod_{e \ni v} f(e) \), \( \forall v \in V \) is a proper vertex coloring. This variant was introduced by Skowronek-Kaziów and it is conjectured that every non-trivial graph \( G \) has an edge-labeling by product, using the labels from \{1, 2, 3\} (Multiplicative 1, 2, 3-Conjecture, see [21]). Currently, we know that every non-trivial graph has an edge-labeling by product, using the labels from \{1, 2, 3, 4\} [21]. Also, every non-trivial, 3-colorable graph \( G \) permits an edge-labeling by product from \{1, 2, 3\} [21]. We will prove that determining whether a given planar 3-colorable graph has an edge-labeling by product from \{1, 2\} is NP-complete.

**(P4) Vertex-labeling by product.**

A vertex-labeling \( f \) is vertex-labeling by product if \( c(v) = \prod_{u \sim v} f(u) \), \( \forall v \in V \) is a proper vertex coloring. For a given graph \( G \), let \{\( V_1, V_2, \cdots, V_k \)\} be the color classes of a proper
vertex coloring of $G$. Label the set of vertices of $V_1$ by 1; also, for each $i$, $1 < i \leq k$ label the set of vertices of $V_i$ by the $(i - 1)$-th prime number; this labeling is a \textit{vertex-labeling by product}. In number theory, the prime number theorem describes the asymptotic distribution of the prime numbers. The prime number theorem implies estimates for the size of the $n$-th prime number $p_n$ (i.e., $p_1 = 2$, $p_2 = 3$, etc.): up to a bounded factor, $p_n$ grows like $n \log(n)$. As a consequence of the prime number theorem we have the following bound: $p_n < n \ln n + n \ln \ln n$, for $n \geq 6$ (see [5] p. 233). So, every graph $G$ has a \textit{vertex-labeling by product}, from $\{1, 2, \cdots, \chi(G)\}$. Here, we ask the following question.

\textbf{Problem 1.} \textit{Does every graph $G$ have a vertex-labeling by product, using the labels $\{1, 2, \cdots, \chi(G)\}$?}

We shown that, every planar graph $G$ has a \textit{vertex-labeling by product} from $\{1, 2, \cdots, 5\}$. We will prove that determining whether a given planar 3-colorable graph has a \textit{vertex-labeling by product} from $\{1, 2\}$ is \textbf{NP}-complete. Furthermore, for every $k$, $k \geq 3$ we show that determining whether a given graph has a \textit{vertex-labeling by product} from $\{1, 2, \cdots, k\}$ is \textbf{NP}-complete.

\textbf{(P5) Edge-labeling by gap.}

An edge-labeling $f$ is \textit{edge-labeling by gap} if
\[
c(v) = \begin{cases} 
  f(e) & \text{if } d(v) = 1, \\
  \max_{e \ni v} f(e) - \min_{e \ni v} f(e) & \text{otherwise},
\end{cases}
\]
is a proper vertex coloring. Every graph $G$ has an \textit{edge-labeling by gap} if and only if it has no connected component isomorphic to $K_1$ or $K_2$ (put the different powers of two $(1, 2, \cdots, 2^{|E(G)|-1})$ on the edges of $G$; this labeling is a vertex-labeling by gap). A similar definition was introduced by Tahraoui et al. [22]. They introduced the following variant: Let $G$ be a graph, $k$ be a positive integer and $f$ be a mapping from $E(G)$ to the set $\{1, 2, \cdots, k\}$. For each vertex $v$ of $G$, the label of $v$ is defined as
\[
c(v) = \begin{cases} 
  f(e) & \text{if } d(v) = 1, \\
  \max_{e \ni v} f(e) - \min_{e \ni v} f(e) & \text{otherwise},
\end{cases}
\]
The mapping $f$ is called \textit{gap vertex-distinguishing labeling} if distinct vertices have distinct labels. Such a coloring is called a \textit{gap-$k$-coloring} and is denoted by $\text{gap}(G)$ [22]. It was conjectured that for a connected graph $G$ of order $n$ with $n > 2$, $\text{gap}(G) \in \{n-1, n, n+1\}$ [22]. They purpose study of the variant of the gap coloring problem that distinguishes the adjacent vertices only.

Let $f$ be an \textit{edge-labeling by gap} form $\{1, 2, \cdots, k\}$ for a graph $G$, we have $k \geq \chi(G) - 1$. First, consider the following example.
Remark 1. Every complete graph \( K_n \) of order \( n \) with \( n > 2 \), has an edge-labeling \( f_n \) by gap form \( \{1, 2, \cdots, \chi(K_n) + 1\} \). Suppose that \( K_3 = v_1v_2v_3 \) and let \( f_3 \) be the following function: \( f_3(v_1v_2) = 4, f_3(v_1v_3) = 1 \) and \( f_3(v_2v_3) = 2 \). Define \( f_n \) recursively.

\[
f_n(v_iv_j) = \begin{cases} 
  f_{n-1}(v_iv_j) + 1 & \text{if } i, j < n, \\
  1 & \text{if } i = n \text{ and } j \neq 2, \\
  2 & \text{otherwise},
\end{cases}
\]

Now, we state the following problem:

**Problem 2.** Does every connected graph \( G \) of order \( n \) with \( n > 2 \), have an edge-labeling by gap form \( \{1, 2, \cdots, \chi(G) + 1\} \)?

We will prove that determining whether a given planar bipartite graph has an edge-labeling by gap from \( \{1, 2\} \) is \( \text{NP} \)-complete. Also, we show that for every \( k, k \geq 3 \), it is \( \text{NP} \)-complete to determine whether a given graph has an edge-labeling by gap from \( \{1, 2, \cdots, k\} \).

(P6) Vertex-labeling by gap. A vertex-labeling \( f \) is vertex-labeling by gap if

\[
c(v) = \begin{cases} 
  f(u)_{u \sim v} & \text{if } d(v) = 1, \\
  \max_{u \sim v} f(u) - \min_{u \sim v} f(u) & \text{otherwise},
\end{cases}
\]

is a proper vertex coloring. A graph may lack any vertex-labeling by gap. Here we ask the following:

**Problem 3.** Does there is a polynomial time algorithm to determine whether a given graph has a vertex-labeling by gap?

We show that, for a given bipartite graph \( G \), determining whether \( G \) has a vertex-labeling by gap from \( \{1, 2\} \) is \( \text{NP} \)-complete. Also, we prove that there is a polynomial time algorithm for determining whether a given planar bipartite graph \( G \) has a vertex-labeling by gap from \( \{1, 2\} \). In sharp contrast, it is \( \text{NP} \)-complete to decide whether a given planar 3-colorable graph \( G \) has a vertex-labeling by gap from \( \{1, 2\} \).

Every bipartite graph \( G = [X, Y] \) has a vertex-labeling by gap, label the set of vertices \( X \) by 1 and label the set of vertices of \( Y \) by different powers of two \( (2^1, \cdots, 2^{|Y|}) \). Here we ask the following:

**Problem 4.** Does there is a constant \( k \) such that every bipartite graph \( G \), have a vertex-labeling by gap form \( \{1, 2, \cdots, k\} \)?

4
It was shown by Thomassen [23] that, for any \( k \)-uniform and \( k \)-regular hypergraph \( H \), if \( k \geq 4 \), then \( H \) is 2-colorable. For every \( r \)-regular bipartite graph \( G = [X, Y] \) with \( r \geq 4 \), label the set of vertices of one of the color classes in part \( X \) by 1 and label other vertices by 2. This labeling is a vertex-labeling by gap from \( \{1, 2\} \) for \( G \).

(P7) Vertex-labeling by degree. (Fictional coloring) A vertex-labeling \( f \) is vertex-labeling by degree if \( c(v) = f(v)d(v) \), where \( d(v) \) is the degree of vertex \( v \) is a proper vertex coloring. This parameter was introduced by Bosek, Grytczuk, Matecki and Żelazny [26]. They conjecture that every graph \( G \) has a vertex-labeling by degree from \( \{1, 2, \ldots, \chi(G)\} \). Let \( p \) be a prime number and let \( G \) be a graph such that \( \chi(G) \leq p-1 \), they proved that \( G \) has a vertex-labeling by degree from \( \{1, 2, \ldots, p-1\} \). For every \( k \) greater than two it is clear that determining whether a given graph has a vertex-labeling by degree from \( \{1, 2, \ldots, k\} \) is NP-complete. We will prove that determining whether a given graph has a vertex-labeling by degree from \( \{1, 2\} \) is in \( P \).

(P8) Vertex-labeling by maximum. A vertex-labeling \( f \) is vertex-labeling by maximum if \( c(v) = \max_{u \sim v} f(u) \), \( \forall v \in V \) is a proper vertex coloring. A graph \( G \) may lack any vertex-labeling by maximum and it has a vertex-labeling by maximum from \( \{1, 2\} \) if and only if \( G \) is bipartite. We present a nontrivial necessary condition that can be checked in polynomial time for a graph to have a vertex-labeling by maximum.

Remark 2 Let \( k \) be the minimum number such that \( G \) has a vertex-labeling by maximum from the set \( \{1, 2, \ldots, k\} \), then \( \chi(G) - k \) can be arbitrary large. For instance, for a given \( t > 3 \) consider the graph \( G \) with vertex set \( V(G) = \{a_i : 1 \leq i \leq t\} \cup \{b_j : 1 \leq j \leq t-2\} \) and edge set \( E(G) = \{a_ia_{i+1} : 1 \leq i \leq t-1\} \cup \{a_jb_j, b_ja_{j+1} : 1 \leq j \leq t-2\} \). Clearly \( k - \chi(G) = t - 3 \).

We will show that determining whether a given 3-regular graph has a vertex-labeling by maximum from \( \{1, 2, 3\} \) is NP-complete.

Throughout this paper all graphs are finite and simple. We follow [13, 25] for terminology and notation not defined here, and we consider finite undirected simple graphs \( G = (V, E) \). We denote the induced subgraph \( G \) on \( S \) by \( G[S] \). Also, for every \( v \in V(G) \) and \( S \subseteq V(G) \), \( N(v) \) and \( N(S) \) denote the neighbor set of \( v \) and the set of vertices of \( G \) which has a neighbor in \( S \), respectively. A proper vertex coloring of \( G = (V, E) \) is a function \( c : V(G) \rightarrow L \), such that if \( u, v \in V(G) \) are adjacent, then \( c(u) \) and \( c(v) \) are different. A proper vertex \( k \)-coloring is a proper vertex coloring with \( |L| = k \). The smallest integer \( k \) such that \( G \) has a proper vertex \( k \)-coloring is called the chromatic number of \( G \).
Table 1: Graph Labeling Results

<table>
<thead>
<tr>
<th>Edge-labeling by</th>
<th>{1, 2}</th>
<th>{1, 2, 3}</th>
<th>Current Upper Bound</th>
<th>Conjecture</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sum</td>
<td>NP-c</td>
<td>-</td>
<td>{1, 2, 3, 4, 5}</td>
<td>{1, 2, 3}</td>
</tr>
<tr>
<td>Product</td>
<td>NP-c</td>
<td>-</td>
<td>{1, 2, 3}</td>
<td>{1, 2, 3}</td>
</tr>
<tr>
<td>Gap</td>
<td>NP-c</td>
<td>NP-c</td>
<td>{1, 2, \ldots, 2^{</td>
<td>E(G)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Vertex-labeling by</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Sum</td>
<td>NP-c</td>
<td>NP-c</td>
<td>\Delta^2 - \Delta + 1</td>
<td>{1, 2, \ldots, \chi}</td>
</tr>
<tr>
<td>Product</td>
<td>NP-c</td>
<td>NP-c</td>
<td>{1, \ldots, \chi \ln \chi + \chi \ln \ln \chi + 2}</td>
<td>{1, 2, \ldots, \chi}</td>
</tr>
<tr>
<td>Degree</td>
<td>P</td>
<td>NP-c</td>
<td>{1, 2, \ldots, 2\chi}</td>
<td>{1, 2, \ldots, \chi}</td>
</tr>
<tr>
<td>Maximum</td>
<td>P</td>
<td>NP-c</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Gap</td>
<td>NP-c</td>
<td>NP-c</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

and denoted by $\chi(G)$. Similarly, for $k \in \mathbb{N}$, a proper edge $k$-coloring of $G$ is a function $c : E(G) \to \{1, \ldots, k\}$, such that if $e, e' \in E(G)$ share a common endpoint, then $c(e)$ and $c(e')$ are different. The smallest integer $k$ such that $G$ has a proper edge $k$-coloring is called the edge chromatic number of $G$ and denoted by $\chi'(G)$. By Vizing’s theorem [24], the edge chromatic number of a graph $G$ is equal to either $\Delta(G)$ or $\Delta(G) + 1$. Those graphs $G$ for which $\chi'(G) = \Delta(G)$ are said to belong to Class 1, and the others to Class 2.

2 Results

2.1 Edge-labeling by product

**Theorem 1** For a given planar 3-colorable graph $G$, determining whether $G$ has an edge-labeling by product from $\{1, 2\}$ is NP-complete.

**Proof** Clearly, the problem is in NP. We reduced Cubic Planar 1-In-3 3-Sat to our problem. Moore and Robson [18] proved that the following problem is NP-complete.

Cubic Planar 1-In-3 3-Sat.

**INSTANCE:** Set $X$ of variables, collection $C$ of clauses over $X$ such that each clause $c \in C$ has $|c| = 3$ and every variable appears in exactly three clauses, there is no negation in the formula, and the bipartite graph obtained by linking a variable and a clause if and only if the variable appears in the clause, is planar.

**QUESTION:** Is there a truth assignment for $X$ such that each clause in $C$ has exactly one
true literal?

Figure 1: The two gadgets $H_x$ and $I_c$. $I_c$ is on the left hand side of the figure.

Consider an instance $\Phi$, we transform this into a graph $G_\Phi$ such that $G_\Phi$ has an edge-labeling by product from \{1, 2\} if and only if $\Phi$ has a 1-in-3 assignment. We use two gadgets $H_x$ and $I_c$ which are shown in Figure 1. The graph $G_\Phi$ has a copy of $H_x$ for each variable $x \in X$ and a copy of $I_c$ for each clause $c \in C$. Also, for each clause $c = y \lor z \lor w$ add the edges $cy$, $cz$ and $cw$. First, suppose that $G_\Phi$ has an edge-labeling by product from \{1, 2\}. In every copy of $H_x$ and $I_c$ the label of every edge is determined uniquely. See Figure 1 (the label of each edge is written on the edge and the color of each vertex induced by edge labels is written on the vertex). Every variable $x$ appears in exactly three clauses, suppose that $x$ appears in $c_i$, $c_j$ and $c_k$. By attention to the structure of $H_x$ the set of labels of edges $c_i x$, $c_j x$ and $c_k x$ are \{1, 1, 1\} or \{2, 2, 2\}. Furthermore, by attention to the $H_x$ and $I_c$, for every clause $c = x \lor y \lor z$, the set of labels of edges $cx$, $cy$ and $cz$ is \{2, 1, 1\}. Now, for every variable $x$, which is appeared in $c_i$, $c_j$ and $c_k$ put $\Gamma(x) = True$ if and only if the set of labels of edges $c_i x$, $c_j x$ and $c_k x$ is \{2, 2, 2\}. Clearly, $\Gamma$ is an 1-in-3 satisfying assignment. Next, suppose that $\Phi$ has an 1-in-3 satisfying assignment $\Gamma : X \rightarrow \{true, false\}$, for every variable $x$, which is appeared in $c_i$, $c_j$ and $c_k$, label $c_i x$, $c_j x$ and $c_k x$ by 2 if and only if $\Gamma(x) = True$. The labels of other vertices are determined uniquely and it is clear the this labeling is an edge-labeling by product from \{1, 2\}.

2.2 Vertex-labeling by product

In the next, we consider the computational complexity of vertex-labeling by product.
Theorem 2 For a given planar 3-colorable graph \( G \), determining whether \( G \) has a vertex-labeling by product from \( \{1, 2\} \) is NP-complete.

Proof Clearly, the problem is in NP. We reduced Cubic Planar 1-In-3 3-Sat to our problem. First, we construct an auxiliary graph \( H_c \). Put a copy of triangle \( K_3 = z_1^c z_2^c z_3^c \). For every vertex \( z_j^c \), 1 \( \leq \) \( j \) \( \leq \) 2, put 2\( i \) new isolated vertices \( t_{i1}^c, t_{i2}^c, \ldots, t_{i2i}^c \) and join \( z_j^c \) to all of them. Also, add the edges \( t_{11}^c t_{21}^c, t_{12}^c t_{22}^c, \ldots, t_{2i-1}^c t_{2i}^c \). Next, put 2\( i \) new isolated vertices \( t_{1i}^c, t_{2i}^c, \ldots, t_{2i-2i}^c \) and join \( z_j^c \) to all of them. Finally, add the edges \( t_{1i}^c t_{1i}^c, t_{12}^c t_{3i}^c, \ldots, t_{2i-2i}^c t_{2i-2i}^c \).

Call the resulting graph \( H_c \). Now, consider an instance \( \Psi \), we transform this into a graph \( G_\Psi \) such that \( G_\Psi \) has a vertex-labeling by product from \( \{1, 2\} \) if and only if \( \Psi \) has a 1-in-3 assignment. Our construction consists of three steps.

Step 1. For each clause \( c \in C \) put a vertex \( c \) and a copy of \( H_3^c, H_5^c \) and \( H_6^c \). Connect the vertex \( z_j^c \) of \( H_3^c \) to \( c \), also, join the vertex \( z_j^c \) of \( H_5^c \) to \( c \) and finally, connect the vertex \( z_j^c \) of \( H_6^c \) to \( c \).

Step 2. For each variable \( x \in X \) put a vertex \( x \).

Step 3. For each clause \( c = x \lor y \lor w \) add the edges \( cx, cy \) and \( cw \).

First, suppose that \( G_\Psi \) has a vertex-labeling \( f \) by product from \( \{1, 2\} \) and let \( \ell \) be the induced coloring by \( f \). In every copy of \( H_3^c \) the label of vertex \( z_j^c \) is 2. We have the similar property for \( H_5^c \) and \( H_6^c \). By attention to the structure of \( H_3^c \), we have \( f(c) = 1 \) and \( \ell(z_j^c) = 8 \); similarly for \( H_5^c \), we have \( \ell(z_j^c) = 32 \) and for \( H_6^c \), we have \( \ell(z_j^c) = 64 \). So for every clause vertex \( c \) we have \( \ell(c) = 16 \). Now, for every variable \( x \), put \( \Gamma(x) = \text{True} \) if and only if \( f(x) = 2 \). Since for every clause \( c \), \( \ell(c) = 16 \), \( \Gamma \) is an 1-in-3 satisfying assignment. Next, suppose that \( \Psi \) is 1-in-3 satisfiable with the satisfying assignment \( \Gamma : X \rightarrow \{\text{true, false}\} \), for every variable \( x \), label the vertex \( x \) by 2 if and only if \( \Gamma(x) = \text{True} \). The labels of other vertices are determined uniquely and it is clear the this labeling is a vertex-labeling by product from \( \{1, 2\} \).

\( \square \)

Theorem 3 For every \( k, k \geq 3 \), it is NP-complete to determine whether a given graph has a vertex-labeling by product from \( \{1, 2, \ldots, k\} \).

Proof We present a polynomial time reduction from 3-colorability to our problem.

3-Colorability. Given a graph \( G \); is \( \chi(G) \leq 3 \)?

First define the following sets: \( A_k = \{mn : m, n \in \mathbb{N}_k\} \), \( B_k = \{m/n : m, n \in \mathbb{N}_k\} \), where \( \mathbb{N}_k = \{1, 2, \ldots, k\} \). Also, define \( \alpha(k) = \max_{D_k \in C_k} |D_k| \), where \( C_k \) is the set of sets such that for every set \( D_k \in C_k \), we have \( D_k \subseteq A_k \) and \( \{d, d' \in D_k \} \cap B_k = \emptyset \). Since \( k \) is constant, so we can compute \( \alpha(k) \) in \( O(1) \). Now, for a given graph \( G \) with \( n \)
vertices \( v_1, v_2, \ldots, v_n \), join all vertices of \( G \) to the all vertices of complete graph \( K_{\alpha(k)-3} \) with vertices \( v_{n+1}, \ldots, v_{n+\alpha(k)-3} \). Call the resulting graph \( G^* \). Now consider the graph \( G^{**} \) with the vertex set \( \{v^j_i : i \in \mathbb{N}_{n+\alpha(k)-3}, j \in \mathbb{N}_k\} \) such that \( v^j_i \) is joined to \( v^w_x \) if and only if \( x = z \) or \( v_x v_z \in E(G^*) \). Finally, consider a copy of graph \( G^{**} \), for every \( i, 1 \leq i \leq n+\alpha(k)-3 \), put two new isolated vertices \( v'_i \) and \( v''_i \) and join them to the set of vertices \( \{v^1_i, \ldots, v^k_i\} \). Call the resulting graph \( \tilde{G} \) (see Figure 2). We show that \( \tilde{G} \) has a vertex-labeling by product from \( \{1, 2, \ldots, k\} \) if and only if \( G \) is 3-colorable. Let \( f \) be a vertex-labeling by product for \( \tilde{G} \). Clearly, \( f(v^1_i), \ldots, f(v^k_i) \) should be different numbers. For every \( i, i \in \mathbb{N}_{n+\alpha(k)-3} \), we have: \( \{f(v^j_i) : j \in \mathbb{N}_k\} = \mathbb{N}_k \). Furthermore, for every \( i_1, i_2, 1 \leq i_1 < i_2 \leq n+\alpha(k)-3 \), we have: \( f(v'^{i_1}_{i_1}), f(v'^{i_2}_{i_1}), f(v'^{i_1}_{i_2}), f(v'^{i_2}_{i_2}) \in A_k \). Also, for every \( i_1 \) and \( i_2 \), if \( v_{i_1} v_{i_2} \in E(G) \), then

\[
\frac{f(v'^{i_1}_{i_1})f(v'^{i_2}_{i_2})}{f(v'^{i_2}_{i_1})f(v'^{i_2}_{i_2})} \notin B_k.
\]

Therefore, \( \{f(v'^{i}_{i}) : 1 \leq i \leq n+\alpha(k)-3\} \geq \alpha(k) - 3 + \chi(G) \). So, \( \tilde{G} \) has a vertex-labeling by product from \( \{1, 2, \ldots, k\} \) if and only if \( \chi(G) \leq 3 \). The proof is complete.
2.3 Edge-labeling by gap

Theorem 4 For a given planar bipartite graph $G$, determining whether $G$ has an edge-labeling by gap from $\{1,2\}$ is NP-complete.

Proof Let $\Phi$ be a 3-SAT formula with clauses $C = \{c_1, \ldots, c_k\}$ and variables $X = \{x_1, \ldots, x_n\}$. Let $G(\Phi)$ be a graph with the vertices $C \cup X \cup (\neg X)$, where $\neg X = \{-x_1, \ldots, -x_n\}$, such that for each clause $c_j = y \lor z \lor w$, $c_j$ is adjacent to $y, z$ and $w$, also every $x_i \in X$ is adjacent to $\neg x_i$. $\Phi$ is called planar 3-SAT type 2 formula if $G(\Phi)$ is a planar graph. It was shown that the problem of satisfiability of planar 3-SAT type 2 is NP-complete [11].

Planar 3-SAT type 2.

Instance: A 3-SAT type 2 formula $\Phi$.

Question: Is there a truth assignment for $\Phi$ that satisfies all the clauses?

We reduce planar 3-SAT type 2 problem to our problem. In planar 3-SAT type 2, if we only consider the set of formulas such that the bipartite graph $G$ obtained by linking a variable and a clause if and only if the variable appears in the clause, is connected and it does not have any vertex of degree one, the problem remains NP-complete. We reduce this version to our problem. Consider an instance $\Phi$, we transform this into a graph $G_\Phi$ such that $G_\Phi$ has an edge-labeling by gap from $\{1,2\}$ if and only if $\Phi$ has a satisfying assignment. For each variable $x \in X$ put a copy of path $P_3 = xtx_\neg x$, also, for each clause $c \in C$ put a copy of gadget $P_4 = cc'd'e''m$. Now, put a copy $C_6$. Also, for each clause $c = y \lor z \lor w$ add the edges $cy, cz$ and $cw$. Finally, let $x$ be an arbitrary literal, connect $x$ to one of the vertices of $C_6$. $G_\Phi$ is connected, bipartite and planar. First, suppose that $G_\Phi$ has an edge-labeling $f$ by gap from $\{1,2\}$ and $l$ is the induced proper coloring by $f$. Since for every variable $x$ the degrees of vertices $x$ and $\neg x$ are greater than one, also for every clause $c$ the degree of vertex $c$ is 4 and $G_\Phi$ is connected, hence in the induced coloring $l$ by $f$, for the set of variables $\{x_1, \ldots, x_n\}$ and the set of clauses $\{c_1, \ldots, c_m\}$ we have $l(x_1) = l(\neg x_1) = \cdots = l(x_n) \neq l(c_1) = l(\neg c_1) = \cdots = l(\neg c_m)$ and $l(x_1) \neq 2 \neq l(c_1)$. First, suppose that $l(x) = 1$. Since $x$ is adjacent to one of the vertices of $C_6$, in this situation $G_\Phi$ does not have any edge-labeling $f$ by gap from $\{1,2\}$. So $l(x) = 0$ and $l(c) = 1$. Hence, the labels of all edges incident with $x_1$ are same. Also, for every variable $x$, because of $t_x$, the labels of all edges incident with $x$ are different from the labels of all edges incident with $\neg x$. Now, for every variable $x$, which is appeared in $c_i, c_j, \cdots, c_k$ put $\Gamma(x) = True$ if and only if the labels of edge $c_i x$ is 2. For every clause $c = x \lor y \lor w$, $l(c) = 1$, if the set of labels of edges $\{cx, cy, cw\}$ is $\{1\}$, then since $l(c) = 1$ and by attention to the gadget $cc'd'e''m$, $G$ does not have any edge-labeling $f$ by gap from $\{1,2\}$. So, $2 \in \{f(cx), f(cy), f(cw)\}$.

Therefore, $\Gamma$ is an satisfying assignment. Now, let $\Gamma$ be an satisfying assignment for $\Phi$. 

10
For every variable \( x \), label all the edges incident with \( x \) by 2 if and only if \( \Gamma(x) = \text{True} \). It is easy to extend this labeling to an edge-labeling \( f \) by gap from \( \{1, 2\} \). This completes the proof. \( \square \)

**Theorem 5** For every \( k, k \geq 3 \), it is \( \text{NP} \)-complete to determine whether a given graph has an edge-labeling by gap from \( \{1, 2, \cdots, k\} \).

**Proof** We present a polynomial time reduction from \( k \)-colorability, to our problem.

\( k \)-Colorability: Given a graph \( G \); is \( \chi(G) \leq k \)?

For a given graph \( G \), we construct a graph \( G^* \) such that \( \chi(G) \leq k \) if and only if \( G^* \) has an edge-labeling by gap from \( \{1, 2, \cdots, k\} \). Let \( G \) be a graph, for every vertex \( v \in V(G) \), put a copy \( P_3 = vv'v'' \) and join \( v \) to \( u \) if and only if \( uv \in E(G) \). Call the resulting \( G^* \). First, suppose that \( G^* \) has an edge-labeling \( f \) by gap from \( \{1, 2, \cdots, k\} \) and \( \ell \) is the induced coloring by \( f \). for every vertex \( v \), \( v \in V(G^*) \) of degree more than one, we have \( \ell(v) \in \{0, 1, \cdots, k - 1\} \), so \( \ell \) is also a proper vertex coloring for \( G \). Now, let \( c \) be a proper vertex coloring for \( G \). For every vertex \( v \) in \( V(G^*) \), label all edges incident with \( v \) except \( vv' \) by 1 and label \( vv' \) by \( c(v) \). Finally for every edge \( v'v'' \), label \( v'v'' \) by 1 if \( c(v) \neq 1 \), otherwise label \( v'v'' \) by \( k \). This labeling is an edge-labeling by gap from \( \{1, 2, \cdots, k\} \). \( \square \)

### 2.4 Vertex-labeling by gap

**Theorem 6** For a given bipartite graph \( G \), determining whether \( G \) has a vertex-labeling by gap from \( \{1, 2\} \) is \( \text{NP} \)-complete.

**Proof** We reduce Not-All-Equal-3-Sat to our problem in polynomial time. It is shown that the following problem is \( \text{NP} \)-complete [13].

**Not-All-Equal-3-Sat**

**Instance:** Set \( X \) of variables, collection \( C \) of clauses over \( X \) such that each clause \( c \in C \) has \( |c| = 3 \).

**Question:** Is there a truth assignment for \( X \) such that each clause in \( C \) has at least one true literal and at least one false literal?

For a given \( \Phi \), we transform \( \Phi \) into a graph \( G_\Phi \) such that \( G_\Phi \) has a vertex-labeling by gap from \( \{1, 2\} \) if and only if \( \Phi \) has a satisfying assignment. Construction of \( G_\Phi \) is similar to the proof Theorem 4, except the gadget \( P_4 = cc'c''c''' \). For each clause \( c \in C \) instead of \( P_4 = cc'c''c''' \), put a isolated vertex \( c \). First, suppose that \( G_\Phi \) has an edge-labeling \( f \) by gap from \( \{1, 2\} \) and \( l \) is the induced proper coloring by \( f \). By an argument similar to argument of proof of Theorem 4, for every clause \( c = x \lor y \lor w, l(c) = 1 \). So
\( \{f(x), f(y), f(w)\} = \{1, 2\} \), therefore \( \Gamma \) is a NAE satisfying assignment. Now, let \( \Gamma \) be an satisfying assignment for \( \Phi \). For every variable \( x \), label the vertex \( x \) by 2 if and only if \( \Gamma(x) = True \). This completes the proof.

\[ \square \]

**Theorem 7** For a given planar bipartite graph \( G \), determining whether \( G \) has a vertex-labeling by gap from \( \{1, 2\} \) is in \( \mathbf{P} \).

**Proof** First we show that every tree \( T \) with more than two vertex has a vertex-labeling by gap from \( \{1, 2\} \). Let \( T \) be a tree with more than two vertex and \( v \in V(T) \) be an arbitrary vertex, define:

\[
f(u) = \begin{cases} 
1 & \text{if } d(u,v) \equiv 0 \pmod{4}, \\
2 & \text{otherwise},
\end{cases}
\]

We call this kind of labeling as good labeling by center \( v \). It is easy to see that good labeling by center \( v \) is a vertex-labeling by gap from \( \{1, 2\} \). Now, consider the following problem.

**Planar Not-All-Equal 3-Sat.**

**Instance**: Set \( X \) of variables, collection \( C \) of clauses over \( X \) such that each clause \( c \in C \) has \( |c| = 3 \) and the following graph obtained from 3-Sat is planar. The graph has one vertex for each variable, one vertex for each clause; all variable vertices are connected in a simple cycle and each clause vertex is connected by an edge to variable vertices corresponding to the literals present in the clause.

**Question**: Is there a Not-All-Equal truth assignment for \( X \)?

It was proved in [19] that Planar Not-All-Equal 3-Sat is in \( \mathbf{P} \) by a reduction to a known problem in \( \mathbf{P} \), namely Planar(Simple) MaxCut. By a simple argument it was shown that the following problem is in \( \mathbf{P} \) (for more information see [10]).

**Planar Not-All-Equal 3-Sat Type 2.**

**Instance**: Set \( X \) of variables, collection \( C \) of clauses over \( X \) such that each clause \( c \in C \) has \( |c| \geq 3 \) and the bipartite graph obtained by linking a variable and a clause if and only if the variable appears in the clause, is planar.

**Question**: Is there a Not-All-Equal truth assignment for \( X \)?

Now, consider the following:

**Planar Not-All-Equal Sat Type 2.**

**Instance**: Set \( X \) of variables, collection \( C \) of clauses over \( X \) such that each clause \( c \in C \) has \( |c| \geq 2 \) and the bipartite graph obtained by linking a variable and a clause if and only if the variable appears in the clause, is planar.

**Question**: Is there a Not-All-Equal truth assignment for \( X \)?

We can transform any instance of \( \Phi \) Planar Not-All-Equal Sat Type 2 to an instance \( \Psi \) of Planar Not-All-Equal 3-Sat Type 2 in polynomial time. For a given instance \( \Phi \), for each clause with exactly two literals like \( c = (x \lor y) \), put two clauses \( x \lor y \lor t \) and
Let $G = [X, Y]$ be a planar bipartite graph, remove all vertices of degree one, repeat this procedure to obtain a graph $G' = [X', Y']$ such that $G'$ does not have a vertex of degree one. For every vertex $v \in X'$, consider a variable $v$ in $\Phi$ and for every vertex $u \in Y'$ with $d_G(u) = d_{G'}(u)$ put a clause $(\lor_{v \sim u} v)$ in $\Phi$. Now determine whether $\Phi$ has a Not-All-Equal truth assignment. If $\Phi$ has a Not-All-Equal truth assignment $\Gamma$, for every vertex $v, v \in X'$ label $v$ by $1$ if and only if $\Gamma(v) = \text{False}$. Label other vertices of $G'$ by $2$, call this labeling by $f$. The induced graph on $V(G) \setminus V(G')$ is a forest, call this forest by $F$. Suppose that $F = T_1 \cup \cdots \cup T_k$, where $T_i$ is a tree. For every $i$, $1 \leq i \leq k$ let $v_i$, $v_i \in V(G')$ be a vertex with minimum distance from $T_i$. Now for every $T_i$ four cases can be considered:

Case 1: $v_i \in Y'$ and $\{ \cup_{v \sim u} f(u) \} = \{1, 2\}$. Let $z \in N_{G'}(v_i)$ such that $f(z) = 1$ and $T_i' = T_i \cup v_i \cup z$. Suppose that $f_i$ is a good labeling by center $z$ for $T_i'$.

Case 2: $v_i \in Y'$ and $\{ \cup_{v \sim u} f(u) \} = \{2\}$. Let $z \in N_{T_i'}(v_i)$. Suppose that $f_i$ is a good labeling by center $z$ for $T_i'$.

Case 3: $v_i \in Y'$ and $\{ \cup_{v \sim u} f(u) \} = \{1\}$. Let $z \in N_{G'}(v_i)$ such that $f(z) = 1$ and $T_i' = T_i \cup v_i \cup z$. Suppose that $f_i$ is a good labeling by center $z$ for $T_i'$.

Case 4: $v_i \in X'$ and $\{ \cup_{v \sim u} f(u) \} = \{2\}$. Let $T_i' = T_i \cup v_i \cup t$, where $t$ is anew vertex and $t$ is joined to $v_i$ in $T_i'$. Suppose that $f_i$ is a good labeling by center $t$ for $T_i'$.

It is easy to see that the union of good labelings $f, f_1, f_2, \cdots, f_k$ is a vertex-labeling by gap from $\{1, 2\}$ for $G$. If $\Phi$ does not have a Not-All-Equal truth assignment. Then, for every vertex $v \in X'$, consider a variable $v$ in $\Psi$ and for every vertex $u \in X'$ with $d_G(u) = d_{G'}(u)$ put a clause $(\lor_{v \sim u} v)$ in $\Psi$. Now determine whether $\Psi$ has a Not-All-Equal truth assignment. If $\Phi$ has a Not-All-Equal truth assignment $\Gamma$ by a similar method we can find vertex-labeling by gap from $\{1, 2\}$ for $G$. Otherwise, $G$ does not have any vertex-labeling by gap from $\{1, 2\}$.

\[\square\]

**Theorem 8** For every $k$, $k \geq 3$, it is $\text{NP}$-complete to determine whether a given graph has a vertex-labeling by gap from $\{1, 2, \cdots, k\}$.

**Proof** The proof is similar to the proof of Theorem 5. \[\square\]
It was shown that 3-colorability of planar 4-regular graphs is NP-complete [9]. So we have the following:

**Theorem 9** It is **NP-complete** to decide whether a given planar 3-colorable graph $G$ has a vertex-labeling by gap from $\{1, 2\}$.

### 2.5 Vertex-labeling by degree

For every $k$ greater than three it is clear that determining whether a given graph has a vertex-labeling by degree from $\{1, 2, \cdots, k\}$ is NP-complete.

**Theorem 10** Determining whether a given graph has a vertex-labeling by degree from $\{1, 2\}$ is in **P**.

**Proof** We reduce our problem to 2-SAT problem in polynomial time.  

2-SAT.  

**Instance:** A 2-SAT formula $\Phi$.  

**Question:** Is there a truth assignment for $\Phi$ that satisfies all the clauses?  

For a given graph $G$ of order $n$ we construct a 2-SAT formula $\Phi$ with $n$ variables $v_1, v_2, \cdots, v_n$ such that $G$ has a vertex-labeling by degree from $\{1, 2\}$ if and only if there is a truth assignment for $\Phi$. For every edge $e = v_i v_j$, if $d(v_i) = d(v_j)$, add the clauses $v_i \lor v_j$ and $\neg v_i \lor \neg v_j$ and if $d(v_i) = 2d(v_j)$, add the clause $v_i \lor \neg v_j$, otherwise if $2d(v_i) = d(v_j)$, add the clause $\neg v_i \lor v_j$. First, suppose that $\Gamma$ is satisfying assignment for $\Phi$. For every vertex $v_i$, label $v_i$ by 2 if and only if $\Gamma(v_i) = true$. It is easy to see that this labeling is a vertex-labeling by degree from $\{1, 2\}$. Next, let $f$ be a vertex-labeling by degree from $\{1, 2\}$, for every variable $v_i$, put $\Gamma(v_i) = true$ if and only if $f(v_i) = 2$. As we know 2-SAT problem is in **P** [13]. This completes the proof.  

### 2.6 Vertex-labeling by maximum

A graph may lack any vertex-labeling by maximum, in the next we consider the complexity of vertex-labeling by maximum; also, we present a necessary condition that can be checked in polynomial time for a graph to have a vertex-labeling by maximum.

**Theorem 11** For a given 3-regular graph $G$, determining whether $G$ has a vertex-labeling by maximum from $\{1, 2, 3\}$ is **NP-complete**.
Figure 3: Transformation in constructing $G'$.

**Proof** Clearly, the problem is in NP. It was shown that it is NP-hard to determine the edge chromatic number of a cubic graph [14]. Let $G$ be a 3-regular graph. We construct a 3-regular graph $G'$ from $G$ such that $G'$ has a vertex-labeling by maximum from $\{1, 2, 3\}$ if and only if $G$ belongs to Class 1. In order to construct $G'$, for every vertex $v \in V(G)$ with the neighbors $x$, $y$ and $z$ consider two disjoint triangles $v_xv_yv_z$ and $v'_xv'_yv'_z$ in $G'$.

Also, for every edge $e \in E(G)$, consider two vertices $e$ and $e'$ in $G'$. Finally, for every edge $e = uv \in E(G)$, join $e$ to $v_u$ and $u_v$; also join $e'$ to $v'_u$ and $u'_v$. Name the constructed graph $G'$ (see Figure 3). Since $G'$ has triangles, so every vertex-labeling by maximum needs at least 3 distinct labels. First suppose that $G'$ has a vertex-labeling $f$ by maximum from $\{1, 2, 3\}$ and let $\ell$ be the induced vertex coloring by $f$. For every vertex $v \in V(G)$ with the neighbors $x$, $y$ and $z$ in $G$, we have $\{\ell(v_x), \ell(v_y), \ell(v_z)\} = \{1, 2, 3\} = \{\ell(v'_x), \ell(v'_y), \ell(v'_z)\}$.

Suppose that there are $u$ and $v$ such that $\ell(v_u) = \ell(v'_u) = 3$, then $f(vu) = f((vu)') = 3$. Since $f$ can not assign 3 to the vertices in a triangle, hence $\ell(vu) = \ell((vu)') = 3$ and this is a contradiction. so we have the following fact:

There are no $u$ and $v$ such that $\ell(v_u) = \ell(v'_u) = 3$ (Fact 1).

Now, consider the following proper 3-edge coloring for $G$: $g : E(G) \rightarrow \{1, 2, 3\}$,

\[
g(uv) = \begin{cases} 
1 & \text{if } f(uv) = 3, \\
2 & \text{if } f((uv)') = 3, \\
3 & \text{otherwise.}
\end{cases}
\]

By Fact 1, $g$ is well-defined and $G$ belongs to Class 1. On the other hand, assume that $g : E(G) \rightarrow \{1, 2, 3\}$ is a proper 3-edge coloring. Define $f : V(G') \rightarrow \{1, 2, 3\}$ such that for every edge $uv \in E(G)$, $f(v_u) = f(v'_u) = 1$, $f(uv) = g(uv)$ and $f((uv)') \equiv g(uv) + 1 \pmod{3}$. It is easy to see that $f$ is a vertex-labeling by maximum. $\square$
For a given graph $G$, put a new vertex $v$ and join it to the all vertices of $G$, next put a new vertex $u$ and join it to $v$. Name the constructed graph $G'$. We can construct $G'$ in polynomial time and $G$ has a vertex-labeling by maximum from $\{1, 2, \ldots, k\}$ if and only if $G'$ has a vertex-labeling by maximum from $\{1, 2, \ldots, k + 1\}$, so we have the following:

**Theorem 12** For every $k \geq 3$, it is NP-complete to decide whether $G$ has a vertex-labeling by maximum from $\{1, 2, \ldots, k\}$ for a given $k$-colorable graph $G$.

Every triangle-free graph has a vertex-labeling by maximum (put different numbers on vertices) and if $G$ is graph such that every vertex appears in some triangles then $G$ does not have vertex-labeling by maximum. Here, we present a nontrivial necessary condition for a graph to have a vertex-labeling by maximum. First consider the following definition.

**Definition 1** For a given graph $G$ the subset $S$ of vertices is called kernel if every $v \in S$ appears in a triangle in $G[S]$ and for every two adjacent vertices $v$ and $u$, where $v \in S$ and $u \in N(S) \setminus S$, there exists a vertex $z \in S$ such that $z$ is adjacent to $v$ and $u$.

Let $S$ be a kernel for $G$. To the contrary, assume that $f$ is a vertex-labeling by maximum for $G$ and $v \in S \cup N(S)$ is a vertex that gets the maximum of $\{f(u) : u \in S \cup N(S)\}$. Then $v$ has two neighbors $x$ and $y$ in $S$ with $\max_{u \sim x} f(u) = \max_{u \sim y} f(u) = f(v)$. This is a contradiction. Therefore, if $G$ has a kernel, then $G$ does not have a vertex-labeling by maximum. Now, consider Algorithm 1.

When Algorithm 1 terminates, if it returns "$G$ has the kernel $S$", then $S$ is a kernel, so $G$ does not have vertex-labeling by maximum. Suppose that Algorithm 1 returns "$G$ has no kernel", but $G$ has a kernel $S'$. In the lines 2 – 3 of algorithm, the set of vertices $S'$ are added to $S$. Now, consider the line 5 of algorithm and let $v \in S'$ be the first vertex form the set $S'$ that is eliminated from $S$. When Algorithm 1 chooses the vertex $v$, $v$ is in a triangle in $G[S']$, so is in a triangle in $G[S]$. Therefore, there is a vertex $u$ such that $uv \in E(G)$, $v \in S'$, $u \in N(S) \setminus S$ and there is no vertex $z \in S$ such that $z$ is adjacent to $v$ and $u$. So $S'$ is not kernel. It is a contradiction. So when Algorithm 1 returns "$G$ has no kernel", $G$ does not have any kernel. Here, we ask the following question: Is the necessary condition, sufficient for a given graph to have a vertex-labeling by maximum?

**Problem 5.** Does there is a polynomial time algorithm to determine whether a given graph has a vertex-labeling by maximum?
Algorithm 1 (Kernel)

\[ S = \emptyset \]
\[ \textbf{for} \ (\text{Every vertex } u \text{ in a triangle}) \ \textbf{do} \]
\[ S \leftarrow S \cup \{u\} \]
\[ \textbf{end for} \]
\[ \textbf{while} \ (\text{There are two adjacent vertices } u \text{ and } v \text{ such that } v \in S, u \in N(S) \setminus S \text{ and there is no vertex } z \in S \text{ such that } z \text{ is adjacent to } v \text{ and } u.) \text{ or } (v \text{ is not in any triangle in } G[S]) \ \textbf{do} \]
\[ S \leftarrow S \setminus \{v\} \]
\[ \textbf{end while} \]
\[ \textbf{if} \ (S \neq \emptyset) \ \textbf{then} \]
\[ \quad \text{Return } "G \text{ has the kernel } S." \]
\[ \textbf{else} \]
\[ \quad \text{Return } "G \text{ has no kernel."} \]
\[ \textbf{end if} \]

3 Acknowledgment

We would like to thank Wiktor Żelazny for his valuable answers to our questions about the definition of fictional coloring.

References


