A 1-2-3-4 result for the 1-2-3 Conjecture in 5-regular graphs

Julien Bensmail

A 1-2-3-4 result for the 1-2-3 Conjecture in 5-regular graphs☆

Julien Bensmail[a]

[a] *Université Côte d'Azur, CNRS, Inria, I3S, France*

Abstract

The 1-2-3 Conjecture, posed by Karoński, Łuczak and Thomason, asks whether every connected graph G different from K_2 can be 3-edge-weighted so that every two adjacent vertices of G get distinct sums of incident weights. Towards that conjecture, the best-known result to date is due to Kalkowski, Karoński and Pfender, who proved that it holds when relaxed to 5-edge-weightings. Their proof builds upon a weighting algorithm designed by Kalkowski for a total version of the problem.

In this work, we present new mechanisms for using Kalkowski's algorithm in the context of the 1-2-3 Conjecture. As a main result we prove that every 5-regular graph admits a 4-edge-weighting that permits to distinguish its adjacent vertices via their incident sums.

Keywords: 1-2-3 Conjecture; four weights; 5-regular graphs.

1. Introduction

In this paper, we deal with variants of the so-called **1-2-3 Conjecture**, which is based on the following definitions and concepts. Let G be an (undirected simple) graph, and let w be an (improper) edge-weighting of G. For each vertex v of G, one can compute its sum

$$\sigma_w(v) := \sum_{u \in N(v)} w(vu)$$

(also denoted $\sigma(v)$ when no ambiguity is possible) of incident weights by w. In case, for this particular weighting w, we get that σ_w is a proper vertex-colouring of G, i.e. no two adjacent vertices of G get the same sum, we call w *sum-colouring*. It can easily be observed that every connected graph different from K_2 admits sum-colouring edge-weightings. Hence, every graph with no connected component isomorphic to K_2 is *nice* regarding the notion of sum-colouring edge-weighting, and it consequently makes sense studying the least number of consecutive weights $1, 2, ..., k$ needed to obtain a such k-edge-weighting for any nice graph. For a given nice graph G, we denote this least k by $\chi^e_\Sigma(G)$.

The 1-2-3 Conjecture, addressed in 2004 by Karoński, Łuczak and Thomason [8], is precisely about the parameter χ^e_Σ. More precisely, it states that $\chi^e_\Sigma(G)$ should be bounded above by 3 for every nice graph G.

1-2-3 Conjecture (Karoński, Łuczak, Thomason)**.** *For every nice graph G, we have $\chi^e_\Sigma(G) \leq 3$.*

The introduction of the 1-2-3 Conjecture gave birth to active investigations dedicated both to the original conjecture and variants of it. For more information on that wide

☆The author was supported by PEPS grant POCODIS.

topic, we refer the interested reader to the survey [10] by Seamone. It has to be known, though, that the general bound on χ^e_Σ proposed in the 1-2-3 Conjecture cannot be pushed further down, as observed notably for nice complete graphs. A more remarkable fact is that, actually, deciding whether $\chi^e_\Sigma(G) \leq 2$ holds for a given nice graph G is NP-complete in general, as first proved by Dudek and Wajc [4]. The similar fact was not known to hold when restricted to nice bipartite graphs G, until Thomassen, Wu and Zhang have recently shown, in [11], that those nice bipartite graphs G with $\chi^e_\Sigma(G) = 3$ can be recognized in polynomial time. Many more investigations dedicated to the true importance of the weight 3 for the 1-2-3 Conjecture can be found in the literature, constituting interesting lines of research.

The best known upper bound on χ^e_Σ so far is derived from a bound on another chromatic parameter for a variant of the 1-2-3 Conjecture called the **1-2 Conjecture**, which we recall now. In 2010, Przybyło and Woźniak studied the implications on the 1-2-3 Conjecture of being allowed to locally alter the incident sum at every vertex. This gives rise to the following definitions and terminology. Let G be a graph, and w be a total-weighting of G. As previously, one can compute, for every vertex v of G, its sum of incident weights by w, being, this time,

$$\sigma_w(v) := w(v) + \sum_{u \in N(v)} w(vu)$$

(or denoted, again, simply $\sigma(v)$ when no ambiguity is possible). Again, we say that w is *sum-colouring* if σ_w is a proper vertex-colouring of G, while $\chi^t_\Sigma(G)$ refers to the least number of consecutive weights $1, 2, ..., k$ needed to k-total-weight G in a sum-colouring way. Since $\chi^t_\Sigma(G) \leq \chi^e_\Sigma(G)$ clearly holds for every nice graph G, and $\chi^t_\Sigma(K_2) = 2$, all graphs admit sum-colouring total-weightings; so no notion of niceness is required in this context. We further note that sum-colouring total-weighting a graph is similar to sum-colouring edge-weighting the same graph where each vertex is attached a pendant vertex.

In [9], Przybyło and Woźniak wondered whether, in the total version of the problem, graphs can in general be weighted with less weights than they can in the edge version. As they could not come up with easy counterexamples, they legitimately addressed the following 1-2 Conjecture.

1-2 Conjecture (Przybyło, Woźniak)**.** *For every graph G, we have $\chi^t_\Sigma(G) \leq 2$.*

The latest progress towards the 1-2-3 and 1-2 Conjectures, as well towards other variants, result from the introduction of a brilliant weighting algorithm due to Kalkowski [6]. In its original version, this algorithm was used to show that every graph admits a sum-colouring 3-total-weighting, and even that such a weighting exists if we restrict all vertices to be weighted 1 or 2. Kalkowski thus proved the following, which is close to the 1-2 Conjecture.

Theorem 1.1 (Kalkowski)**.** *For every graph G, we have $\chi^t_\Sigma(G) \leq 3$.*

Modifications of Kalkowski's algorithm permitted to get new upper bounds on variants of χ^t_Σ. Amongst those variants that are the closest to our investigations, let us mention the work of Gao, Wang and Wu [5], who proved, via a slight modification of Kalkowski's ideas, that the 1-2-3 Conjecture is true if we allow the vertices with a same incident sum to induce a forest. More involved modifications of Kalkowski's algorithm allowed Kalkowski, Karoński and Pfender to prove the best known upper bound on χ^e_Σ so far [7].

Theorem 1.2 (Kalkowski, Karoński, Pfender)**.** *For every nice graph G, we have $\chi^e_\Sigma(G) \leq 5$.*

In this paper, we focus on the 1-2-3 Conjecture in 5-regular graphs. One main motivation for this is that nice regular graphs can, intuitively, be considered as being among the most complicated graphs for the 1-2-3 Conjecture. This is because all vertices in a regular graph have the same set of possible sums by an edge-weighting. Due to Theorem 1.2, we know that, for every nice regular graph G, we have $\chi^e_\Sigma(G) \leq 5$. Still in the context of regular graphs, this upper bound can sometimes be reduced further down. As indicated by Seamone in [10], it is known that $\chi^e_\Sigma(G) \leq \chi(G)$ holds whenever $\chi(G) = 2k+1$ (Karoński, Łuczak, Thomason [8]) or $\chi(G) = 4k$ (Duan, Lu, Yu [3]), where χ refers to the usual chromatic number parameter. Hence, since the 1-2-3 Conjecture holds for nice complete graphs and cycles, the upper bound 5 can be reduced down to 3 for 3-regular graphs, and down to 4 for 4-regular graphs. The 1-2-3 Conjecture was also verified for regular graphs with large degree, see [1, 2].

Our goal in this paper is to improve the bound on the parameter χ^e_Σ for 5-regular graphs (which are obviously all nice), by showing that their index is bounded above by 4.

Theorem 1.3. *For every 5-regular graph G, we have $\chi^e_\Sigma(G) \leq 4$.*

In light of the previous explanations, one first point for considering 5-regular graphs is that they form, in a sense, the class of regular graphs with smallest degree for which the upper bound of 4 on χ^e_Σ is not known to hold. Another more important point lies in the method we use. Namely, to prove Theorem 1.3, we introduce another modification of Kalkowski's algorithm that is rather different from those designed so far. We believe this is of interest, as Kalkowski's algorithm remains, to date, one of the main methods used to deal with the 1-2-3 Conjecture. Lastly, although one may regard 5-regular graphs as a rather restricted class of graphs, our method actually also applies to less natural classes of graphs. We also believe that our method could potentially be generalized to regular graphs with larger degree. These points will be discussed further in the concluding section.

This paper is organized as follows. As our proof of Theorem 1.3 highly relies on Kalkowski's algorithm from [6], we first recall it in Section 2. We voluntarily do so by employing a modified terminology that suits our needs better. Our proof of Theorem 1.3 is then given in Section 3.

2. Kalkowski's algorithm

The proof of Theorem 1.1 by Kalkowski in [6] relies on the fact that every graph admits a 3-edge-weighting which is almost sum-colouring, in the following sense.

Lemma 2.1 (Kalkowski)**.** *For every graph G, there is a proper vertex-colouring $\phi : V(G) \to \mathbb{N}^*$ such that G admits a 3-edge-weighting w verifying*

$$\sigma(v) \in \Phi(v) := (\phi(v) - 1, \phi(v))$$

for every vertex v of G.

The proof of Theorem 1.1 essentially consists in 1) deducing a 3-edge-weighting w of G as guaranteed by Lemma 2.1, then 2) assigning weight 1 to every vertex v verifying $\sigma_w(v) = \phi(v)$, and 3) assigning weight 2 to every vertex v verifying $\sigma_w(v) = \phi(v) - 1$. This results in a sum-colouring 3-total-weighting of G as, for every vertex v, the obtained incident sum is $\phi(v) + 1$ with ϕ being a proper vertex-colouring of G.

Our proof of Theorem 1.3 being based on a refinement of Kalkowski's algorithmic proof of Theorem 1.1, we recall Kalkowski's arguments (using our terminology and notation).

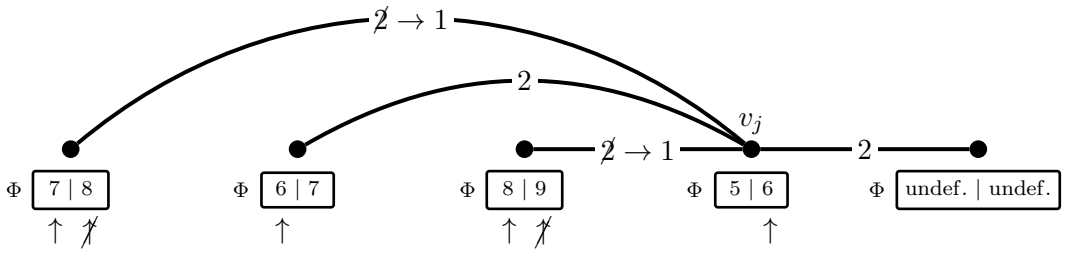

Figure 1: Performing valid adjustments in the proof of Lemma 2.1. Vertex v_j has initial incident sum 8, while its backward neighbours will eventually have final incident sums $7, 8, 9$. We perform valid adjustments backward so that $\sigma(v_j) = 6$, which is an available final incident sum for v_j. Then we set $\Phi(v_j) = (5, 6)$.

Proof of Lemma 2.1. We may assume that G is connected as otherwise we may argue component-wise. Let $v_1, ..., v_n$ be the vertices of G ordered in an arbitrary way. The original proof of Kalkowski, which is purely algorithmic, consists in starting from an original 3-edge-weighting w of G, then processing the v_i's one after another, following the order over their indexes, without coming back at any point, and, whenever treating a new vertex v_i, modifying the weights incident to v_i so that $\Phi(v_i)$ can be chosen conveniently, and $\sigma(v_i)$ belongs to $\Phi(v_i)$. In other words, the $\Phi(v_i)$'s are determined on the fly, while w is being modified at any step to guarantee its existence. Hence, once the algorithm is over, both Φ and w are obtained.

More precisely, the algorithm goes as follows. We start from w assigning weight 2 to every edge of G, and from $\phi(v_i)$ (and thus $\Phi(v_i)$) being undefined for every v_i. The algorithm will respect, during its course, i.e. at every step, a certain number of properties, which are the following:

1. For every already-treated vertex v_i, the pair $\Phi(v_i) := (\phi(v_i) - 1, \phi(v_i))$ is defined, i.e. $\phi(v_i)$ was chosen and we have $\sigma(v_i) \in \Phi(v_i)$.

2. For every two already-treated adjacent vertices v_i and v_j, we have $\phi(v_i) \neq \phi(v_j)$.

3. For every edge v_iv_j with $i < j$, the weight $w(v_iv_j)$ can only be modified when treating v_j.

We note that Property 2 allows $\Phi(v_i) \cap \Phi(v_j)$ to be non-empty, provided $\phi(v_i)$ and $\phi(v_j)$ are different. Furthermore, Property 3 implies 1) that every edge weight is modified at most once during the algorithm's course, and 2) that, whenever treating a new vertex v_j, all backward edges incident to v_j, i.e. those edges of the form v_iv_j with $i < j$, are still assigned weight 2 by w.

We now describe the general behaviour of the algorithm (see Figure 1 for an illustration). Assume all vertices $v_1, ..., v_{j-1}$ have already been treated during the algorithm's course, with Properties 1 to 3 being maintained, and that the next vertex, v_j, is considered (we have $j = 0$ at the very first step). Let $b \leq d(v_j)$ denote the number of backward neighbours of v_j (i.e. vertices v_i, with $i < j$, neighbouring v_j), and arbitrarily denote these vertices by $u_1, ..., u_b$. As said above, remind that we have $w(u_1v_j) = ... = w(u_bv_j) = 2$ at this point of the algorithm. In order to define $\phi(v_j)$, and so $\Phi(v_j)$, with maintaining Property 1, and so that v_j itself satisfies Properties 1 and 2 (once it is treated), we will alter some of the weights of the backward edges incident to v_j. Note that we have to be careful, as, when doing so, one of the u_i's may not fulfil the second part of Property 1 any more. However, since, for every u_i, we have $\sigma(u_i) \in \Phi(u_i)$ and $w(u_iv_j) = 2$, we note

that the weight 2 on u_iv_j can be either incremented or decremented with preserving the fact that $\sigma(u_i) \in \Phi(u_i)$. We call a *valid adjustment* the operation of changing the value of $w(u_iv_j)$ by applying the correct one of these two operations. Hence, by performing valid adjustments to the backward edges incident to v_j, we can modify $\sigma(v_j)$ without having any of the u_i's violating Property 1.

We hence just have to show that there is a set of valid adjustments to the backward edges incident to v_j which makes $\sigma(v_j)$ belonging to $\Phi(v_j)$, so that Property 1 is fully met, for some $\Phi(v_j) = (\phi(v_j)-1, \phi(v_j))$ verifying Property 2. As our proof of Theorem 1.3 partly depends on the existence of such valid adjustments, we prove their existence formally in the following proposition.

Proposition 2.2. *Assume all of vertices $u_1, ..., u_b$ have been treated by the algorithm, i.e. the u_i's verify Property 1, and that v_j is being considered. Then there is a set of valid adjustments to the backward edges incident to v_j for which $\sigma(v_j) \in \Phi(v_j)$, for some $\Phi(v_j) = (\phi(v_j) - 1, \phi(v_j))$ verifying Property 2.*

Proof. When performing a valid adjustment to a backward edge incident to v_j, the value of $\sigma(v_j)$ changes. We only need to show that, by performing some such valid adjustments, we can make $\sigma(v_j)$ or $\sigma(v_j) + 1$ reach a value not among $\{\phi(u_1), ..., \phi(u_b)\}$. Such a value will be our $\phi(v_j)$.

Assume s of the valid adjustments to the backward edges incident to v_j are decrements, while t of the valid adjustments are increments. So we have $b = s + t$. By performing one, two, ..., s decrements, we make $\sigma(v_j)$ decrease by 1, ..., s. Conversely, by performing one, two, ..., t increments, we make $\sigma(v_j)$ increase by 1, ..., t. Hence, by performing some valid adjustments to the backward edges incident to v_j, we can modify $\sigma(v_j)$ to any value among $S := \{\sigma(v_j)-s, ..., \sigma(v_j), ..., \sigma(v_j)+t\}$, which includes $s+t+1 = b+1$ elements. Hence, the set $S \setminus \{\phi(u_1), ..., \phi(u_b)\}$ is non-empty, and we can just choose $\phi(v_j)$ as being any element of this difference, and set $\Phi(v_j) = (\phi(v_j) - 1, \phi(v_j))$. The claimed valid adjustments hence exist. □

Hence, when considering v_j, we can, according to Proposition 2.2, perform valid adjustments to the backward edges incident to v_j yielding a $\Phi(v_j)$ verifying Properties 1 and 2, while the u_i's still verify Property 1. Besides, since valid adjustments concern backward edges of v_j only, Property 3 is still respected. The algorithm can hence pursue its course, hence obtain the claimed edge-weighting, concluding the proof. □

3. Proof of Theorem 1.3

Adapting Kalkowski's algorithm in the context where only the edges are weighted is a tricky task. The strategy proposed, in [7], by Kalkowski, Karoński and Pfender in order to prove Theorem 1.2, relies on several modifications of the algorithm which we describe roughly. First, all $\Phi(v_i)$'s are now of the form $(\phi(v_i) - 2, \phi(v_i))$. Then, since, in the edge version, it is not able to locally adjust a vertex weight to modify the incident sum, it is required, at any point of their algorithm, that $\Phi(v_i) \cap \Phi(v_j)$ is empty for every two adjacent vertices v_i and v_j. Since the latter condition is much stronger than in Kalkowski's original algorithm, an analogue of Proposition 2.2 does not immediately hold. To offset this point, their algorithm is now allowed to adjust the weight of a forward edge (so the ordering $v_1, ..., v_n$ must guarantee that every v_i (but v_n) has a forward neighbour). The price for Kalkowski, Karoński and Pfender's algorithm to work, i.e. to have properties analogous to Properties 1 to 3 to be maintained during its course, is the use of more edge weights.

Our proof of Theorem 1.3 is, essentially, another modification of Kalkowski's algorithm that is, in some sense, closer to the original algorithm than is the approach imagined by Kalkowski, Karoński and Pfender. Instead of just describing a sum-colouring 4-edge-weighting of any 5-regular graph, we below rather describe our approach step by step, so that the reader gets a better intuition on why the resulting weighting is designed in this particular way.

3.1. Rough ideas

The very basic idea behind our proof is to apply Kalkowski's algorithm in the edge context by simulating vertex weights with edge weights. Assume G is a graph we want to edge-weight in a sum-colouring way, and let $W \cup H$ be a partition of $V(G)$ such that every vertex of H has at least one neighbour in W. According to that property, every vertex $u \in H$ has some incident edges going to W. We call those edges the *private edges* of u. We note, now, that a sum-colouring total-weighting w of H naturally yields a partial edge-weighting w' of G which sum-colours the vertices of H only (that is, adjacent vertices of H get different incident sums). One can indeed can just start from w' being w, and then simulate every vertex weight $w(u)$ by setting $w'(uv) = w(u)$, where uv is a private edge of u.

Of course, this idea, as roughly stated above, suffers many issues which need to be pointed out. One issue is that not all edges of G get weighted by w'; this is in particular the case for the private edges not chosen in the last stage. Another issue is that $G[H]$ may consist in several connected components, some of which, in particular the ones with no edges, must be treated differently. Another one main issue is that, by a sum-colouring edge-weighting of G, not only the adjacent vertices of H must receive different sums. In particular, we also have to guarantee that $\sigma(u) \neq \sigma(v)$ for 1) adjacent $u, v \in W$, and 2) adjacent $u \in H$ and $v \in W$. The first of these cases is easy to handle, as we may just require W to be an independent set. For that, we will just make use of the folklore fact that, in any graph, a maximal independent set is also dominating.

Observation 3.1. *Let W be a maximal independent set of a graph G. Then every vertex in $V(G) \setminus W$ has at least one neighbour in W.*

Dealing with the second case above is a bit more complicated, and this is particularly where we will take advantage of the fact that all vertices of G have the same (small) degree. In few words, our edge-weighting will have the property that most edges incident to the vertices in H have "small" weights, namely weights among $\{1, 2, 3\}$, while most edges incident to the vertices in W have "big" weights, namely weights among $\{3, 4\}$. With a careful case analysis, we will make sure that no two adjacent vertices $u \in H$ and $v \in W$ have all their incident edges weighted with weight 3, which is basically the only issue which may occur when the weighting conventions above are fulfilled.

3.2. Definitions and terminology

Let G be a 5-regular graph, which we can assume is connected (otherwise we can argue component-wise). Let W be a maximal independent set of G, and set $H := V(G) \setminus W$. We note that the graph $G[H]$ may consist of several components, which we call *H-components* throughout. In the proof, we need to treat differently those H-components which have edges, and those H-components which consist of an isolated vertex. We call these two types of H-components *good* and *bad*, respectively. For every vertex in a good H-component, there are incident edges going to W. These are precisely the edges we call *private*. For $u \in H$ and $v \in W$ such that u and v are adjacent in G, the edge uv is thus private, and

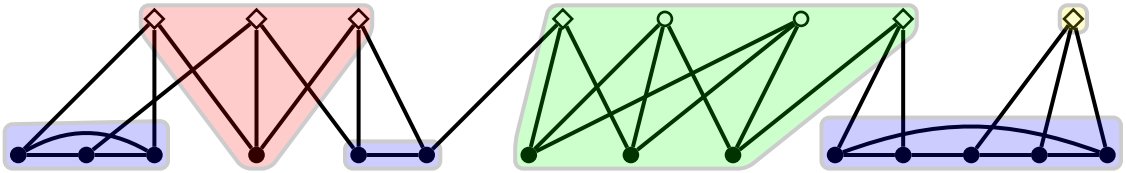

Figure 2: Illustration of the concepts introduced in the proof of Theorem 1.3, for a cubic graph. Black vertices are vertices of H, while white vertices are vertices of W. Diamond vertices of W are external vertices, while circle vertices of W are internal vertices. Good H-components are highlighted in blue. Type-1 components are highlighted in yellow, type-2 components are highlighted in green, while type-3 components are highlighted in red.

we call v a *private neighbour* of u. In particular, the edges incident to the vertex of a bad H-component are not regarded as private. The other way around, a vertex in W might have none to all of its incident edges being private.

Before pursuing, we need to introduce some more terminology (refer to Figure 2 for an illustration of the upcoming concepts). We say that a vertex in W is *internal* if all its neighbours, which are all in H, belong to bad H-components. In particular, an internal vertex of W is not a private neighbour of any vertex in H. All other vertices of W are said *external*, meaning that they are adjacent to vertices in good H-components. In the proof, the edges of G inside/incident to the good H-components will be weighted using a modification of Kalkowski's algorithm. To weight the other edges of G, we will mainly argue depending of the nature of some induced subgraphs of G. Let hence $H := H_{\mathrm{g}} \cup H_{\mathrm{b}}$, where H_{g} and H_{b} denote the vertices of H being in good and bad H-components, respectively. More precisely, we will have to look at the components of $G' := G[H_{\mathrm{b}} \cup W]$. The connected components of G' can be of three main types. First, there are those components of G', which are said to be of *type-1*, which consist of only one isolated vertex (in W). Since W dominates H_{b}, we note that every other component of G' has edges. More precisely, all other components of G' are bipartite graphs in which all vertices in H_{b} have degree exactly 5, while all vertices in W have degree at most 5. In particular, if such a vertex v of W has degree strictly less than 5, then v is external. To conclude the classification of the remaining components of G', we say that such a component is of *type-2* whenever it includes at least one internal vertex, while it is said to be of *type-3* otherwise, i.e. if it includes external vertices only.

3.3. A sum-colouring 4-edge-weighting scheme

We are now ready to describe how to obtain a sum-colouring 4-edge-weighting w of G. This weighting w is mainly obtained though several successive modification stages. The description is done step by step, so that the reader gets aware of all consequences of our weight modifications.

We start by assigning weight 3 to all private edges, i.e. edges uv where $u \in H_{\mathrm{g}}$ and $v \in W$. These weights might be altered later on, so no incident sum is final so far.

We then weight the edges of the good H-components using Kalkowski's algorithm so that, for every two adjacent vertices u and v in a good H-component, we have $\sigma(u) \neq \sigma(v)$. To that aim, the private edges will basically simulate vertex weights. Note that this is possible since, at this point, we have $w(uv) = 3$ for some private edge uv, so for every $u \in H_{\mathrm{g}}$ we have the possibility to increment the incident sum of u by setting $w(uv) = 4$. For every private edge, only this move will potentially be applied, so, eventually, every private edge will be weighted with weight 3 (unchanged) or 4 (incremented).

So that some upcoming arguments are valid, we need to apply Kalkowski's algorithm so that the sums of weights incident to the vertices in H_{g} range in a particular set of small values. Namely, we need the following result:

Lemma 3.2. *Let C be a good H-component of G, and let G_C be the subgraph of G containing C and all private edges of G incident to vertices in C. Then G_C admits a 4-edge-weighting w such that:*

1. *for every $u \in V(C)$, we have $\sigma(u) < 15$,*
2. *for every two adjacent vertices $u, v \in V(C)$, we have $\sigma(u) \neq \sigma(v)$,*
3. *for every private edge uv of G_C, we have $w(uv) \in \{3, 4\}$.*

Proof. We use a proof scheme that is reminiscent of that of Lemma 2.1. Let $v_1, ..., v_n$ denote the vertices of C ordered in an arbitrary way. Recall that all v_i's have degree 5 in G_C, and every v_i is incident to at least one private edge. For every v_i, there are, in G_C, three types of incident edges, namely:

- $b \leq 4$ backward edges (i.e. edges $v_j v_i$ with $j < i$),
- $f \leq 4$ forward edges (i.e. edges $v_i v_j$ with $i < j$),
- $p \geq 1$ private edges (i.e. edges $v_i u$ where $u \in W$).

We start from w assigning weight 3 to all private edges of G_C. We then give an initial weight by w to the edges of C. To that aim, we consider every v_j in turn (and its specific values of b and f), and weight its backward edges as follows:

- in case $f \geq 1$, we set $w(v_i v_j) = 2$ for every edge $v_i v_j$ with $i < j$;
- in case $f = 0$, let $u_1, ..., u_b$ denote the backward neighbours of v_j. Then:
  - if $b = 4$, then we set $w(u_1 v_j) = 1$ and $w(u_2 v_j) = w(u_3 v_j) = w(u_4 v_j) = 2$;
  - if $b = 3$, then we set $w(u_1 v_j) = w(u_2 v_j) = 1$ and $w(u_3 v_j) = 2$;
  - if $b = 2$, then we set $w(u_1 v_j) = w(u_2 v_j) = 1$;
  - if $b = 1$, then we set $w(u_1 v_j) = 1$.

All edges incident to the v_i's in G_C are now assigned an initial weight by w, so each v_i has an initial incident sum $\sigma(v_i)$. Similarly as in Kalkowski's algorithm, we now process the v_i's from first to last, and define, for every v_i, the set $\Phi(v_i) := (\phi(v_i) - 1, \phi(v_i))$ so that $\phi(v_i) \neq \phi(v_j)$ for every two adjacent vertices v_i, v_j of C. We claim that, from the initial 4-edge-weighting of C we have defined, this is possible in such a way that $\phi(v_i) < 15$ (so that the third item of the statement is met).

Consider a vertex v_j such that the previous v_i's have been treated. Recall that v_j has b backward neighbours $u_1, ..., u_b$, f forward neighbours, and $p \geq 1$ private neighbours $w_1, ..., w_p$. Similarly as in the proof of Proposition 2.2, we can make valid adjustments backwards. The difference, though, is that some edges $u_i v_j$ have been weighted 1 and cannot be decremented. More precisely, the valid adjustments backwards are as follows:

- $u_i v_j$ is an edge with $w(u_i v_j) = 2$ and $\sigma(u_i) = \phi(u_i) - 1$: the weight $w(u_i v_j)$ can be incremented;

- u_iv_j is an edge with $w(u_iv_j) = 1$ and $\sigma(u_i) = \phi(u_i) - 1$: the weight $w(u_iv_j)$ can be incremented;

- u_iv_j is an edge with $w(u_iv_j) = 2$ and $\sigma(u_i) = \phi(u_i)$: the weight $w(u_iv_j)$ can be decremented.

We denote by s the number of possible valid decrements backwards, and by t the number of possible valid increments backwards. Clearly, $s + t \leq 4$ since $p \geq 1$. By the remarks above, we might have $s + t < b$ when $f = 0$.

First assume $f \geq 1$. Then all backward edges incident to v_j have weight 2, meaning that $s + t = b$. Then Proposition 2.2 can be applied directly. Furthermore, because $f \geq 1$, there is a sequence of valid adjustments such that $\phi(v_j) \leq 14$. To see this holds, note that, by performing valid adjustments backwards, we can make $\sigma(v_j)$ take any value among $\{\sigma(v_j) - s, ..., \sigma(v_j), ..., \sigma(v_j) + t\}$, meaning that we can choose, as $\phi(v_j)$, any element in

$$\{\sigma(v_j) - s, ..., \sigma(v_j), ..., \sigma(v_j) + t + 1\} \setminus \bigcup_{i=1}^{b} \{\phi(u_i)\},$$

which includes at least two values, one of which is strictly smaller than 15.

Now assume that $f = 0$. Since there are no forward edges incident to v_j, we here do not have to care about the set $\Phi(v_j)$, and we can just perform valid adjustments so that $\sigma(v_j)$ gets different from $\phi(u_i)$ for every backward edge u_iv_j. Remind that some backward edges are here weighted 1, hence that it may be that some backward edges cannot be adjusted. We note, however, that the number of such backward edges is always smaller or equal to p, the number of private edges incident to v_j. So, although we cannot perform b valid adjustments onto the backward edges incident to v_j, we have to take into account that p valid increments can be performed onto the private edges (and these edges do not prevent v_j from having a particular incident sum).

We distinguish the possible values of b:

- If $b = 4$, then $\sigma(v_j)$ is currently equal to 10, and only $\phi(u_1), \phi(u_2), \phi(u_3), \phi(u_4)$ have to be avoided as $\sigma(v_j)$. Since at least four valid adjustments (at least three from the backward edges weighted 2, one from the private edge) can be performed, there is a sequence of at most four valid adjustments which yields a satisfying value of $\sigma(v_j)$. In the worst-case scenario, we have to perform four increments. So there is a correct sequence yielding $\sigma(v_j) \leq 14$.

- If $b = 3$, then $\sigma(v_j)$ is currently equal to 10, and only $\phi(u_1), \phi(u_2), \phi(u_3)$ have to be avoided as $\sigma(v_j)$. Since at least three valid adjustments (at least one from the backward edges weighted 2, two from the private edges) can be performed, there is a sequence of at most three valid adjustments which yields a satisfying value of $\sigma(v_j)$. In the worst-case scenario, we have to perform three increments. So there is a correct sequence yielding $\sigma(v_j) \leq 13$.

- If $b = 2$, then $\sigma(v_j)$ is currently equal to 11, and only $\phi(u_1), \phi(u_2)$ have to be avoided as $\sigma(v_j)$. Since at least two valid adjustments (from the private edges) can be performed, there is a sequence of at most two valid adjustments which yields a satisfying value of $\sigma(v_j)$. In the worst-case scenario, we have to perform two increments. So there is a correct sequence yielding $\sigma(v_j) \leq 13$.

- If $b = 1$, then $\sigma(v_j)$ is currently equal to 13, and only $\phi(u_1)$ has to be avoided as $\sigma(v_j)$. Since at least one valid adjustment (from the private edges) can be performed, there is a sequence of at most one valid adjustment which yields a satisfying value of $\sigma(v_j)$. In the worst-case scenario, we have to perform one increment. So there is a correct sequence yielding $\sigma(v_j) \leq 14$.

We have now modified w so that:

- For every v_i with $f \geq 1$, we have $\sigma(v_i) \in (\phi(v_i) - 1, \phi(v_i))$ and $\phi(v_i) < 15$;
- For every v_i with $f = 0$, we have $\sigma(v_i) < 15$;
- For every edge v_iv_j such that $\phi(v_i), \phi(v_j)$ are defined, we have $\phi(v_i) \neq \phi(v_j)$;
- For every edge v_iv_j such that $\phi(v_j)$ is not defined, we have $\sigma(v_j) \neq \phi(v_i)$;
- All private edges incident to a v_i with $f \geq 1$ are weighted 3.

To finish off the edge-weighting of G_C, we consider every vertex v_i with $f \geq 1$. If $\sigma(v_i) = \phi(v_i)$, then we do nothing. Otherwise, i.e. $\sigma(v_i) = \phi(v_i) - 1$, we consider a private edge v_iu incident to v_i, and set $w(v_iu) = 4$ so that $\sigma(v_i) = \phi(v_i)$. Since $\phi(v_i) \neq \phi(v_j)$ for every edge v_iv_j where v_i and v_j have their f being non-zero, and $\sigma(v_j)$ was chosen so that $\sigma(v_j) \neq \phi(v_i)$ for every edge v_iv_j with v_j having its f being zero, we get that w is sum-colouring. Furthermore, all additional requirements are met. □

Back to the proof of Theorem 1.3, let us now assume that all edges inside/incident to the good H-components of G are weighted as stated in Lemma 3.2. That is, every two adjacent vertices in H_{g} have different incident sums strictly smaller than 15, and every private edge is weighted with weight 3 or 4. Recall that the type-1 components of G are vertices of W whose all incident edges go to H_{g}. Since all private edges are weighted with weight 3 or 4, every vertex v being a type-1 component verifies $\sigma(v) \geq 15$, which is strictly greater than the sum of weights incident to any vertex in H_{g}. Thus, no sum conflict can involve vertex from a type-1 component, and this will remain true since we will not modify the weights assigned to the private edges.

It now remains to weight the edges of the type-2 and type-3 components of G. This will be done in such a way that the majority of the edges incident to the external vertices are weighted 3 or 4, to make sure that almost all these vertices have incident sum at least 15 (so that no sum conflict can involve an external vertex and a vertex of H_{g}). Note that an external vertex belongs to only one type-2 or type-3 component. So we can edge-weight all type-2 and type-3 components one after another. In the next result, we prove that, indeed, we can weight the edges of these components correctly.

Lemma 3.3. *Let $C = (W, H)$ be a type-2 or type-3 component of G, and assume each vertex v of W is associated a possibly null bias $\gamma(v) \geq 3 \cdot (5 - d_C(v))$, and a list of $5 - d_C(v)$ forbidden values $L(v)$ as $\sigma(v) + \gamma(v)$ with value at most 14. Then C admits a 4-edge-weighting w where 1) $\sigma(u) \neq \sigma(v) + \gamma(v)$ for every two adjacent vertices $u \in H$ and $v \in W$, and 2) $\sigma(v) + \gamma(v) \notin L(v)$ for every external vertex $v \in W$.*

Proof. Throughout this proof, when speaking of the incident sum of a vertex $v \in W$, we mean $\sigma(v) + \gamma(v)$ (i.e. with taking the bias into account). To make sure that $\sigma(v) + \gamma(v)$ is not in $L(v)$, we will, most of the time, just ensure that this sum has value at least 15. We start from w assigning weight 4 to all edges of C. If no sum conflict arises, then we are done. Otherwise, it means that pairs of adjacent vertices of C have the same incident sum.

Note that w has the property that $\sigma(u)$ is even for every $u \in H$, while, for any $v \in W$, the parity of $\sigma(v) + \gamma(v)$ depends on the parity of $\gamma(v)$.

Let v_1, v_2 be two distinct vertices of W, and let P be a path of C with end-vertices v_1 and v_2 (which exists by the connectedness of C). It is easy to see that, along P, turning 3's into 4's and vice-versa results in a new 4-edge-weighting where only the parities of the incident sums of v_1 and v_2 have been changed. Using this fact, we note that if W includes an even number of vertices with even bias, then we can modify w by switching 3's and 4's along paths joining pairs of these vertices with even bias, until we get to the point where $\sigma(v) + \gamma(v)$ is odd for every $v \in W$ and $\sigma(u)$ is even for every $v \in H$.

We may thus assume that the number of vertices in W with even bias is odd. If C is a type-3 component, i.e. W includes a vertex v^* with $d_C(v^*) = 5$ (and thus $\gamma(v^*) = 0$), then we proceed as follows. Using similar arguments as above, we can switch 3's and 4's along some paths of C until we get to the point where, by w, the vertex v^* is the only vertex of W for which $\sigma(v^*) + \gamma(v^*)$ is even. Let $u_1, ..., u_5$ denote the five neighbours of v^* in C. Then we decrease each weight $w(v^*u_i)$ by 2. In the resulting w, we note that v^* is the only vertex in W being incident to edges assigned weight 1 or 2, while all $\sigma(u_i)$'s remain even. Furthermore, each one of these sums has value at least $1 + 4 \cdot 3 = 13$, while $\sigma(v^*)$ is at most $2 \cdot 5 = 10$. So w fulfils all required conditions.

We may lastly also suppose that C is a type-2 component, i.e. all vertices of W are external, and thus have positive bias. Because assigning weight 4 to all edges of C created sum conflicts, it means that a vertex $v^* \in W$ has even bias $\gamma(v^*) = 4 \cdot (5 - d_C(v^*))$. So v^* is part of the odd number of vertices in W having even bias. As previously, we switch 3's and 4's along paths of C until we get to the point where, in W, only v^* verifies that $\sigma(v^*) + \gamma(v^*)$ is even. Let $u_1, ..., u_d \in H$ denote the neighbours of v^* in C. To make sure that $\sigma(v^*) + \gamma(v^*)$ is eventually different from the $\sigma(u_i)$'s, we decrease some edge weights depending on the value of $\mathrm{d}_C(v)$:

- If $d_C(v^*) = 1$, then $\gamma(v^*) = 4 \cdot 4 = 16$ and $w(v^*u_1) = 4$ since $\sigma(v^*) + \gamma(v^*)$ is even. Therefore, $\sigma(v^*) = \sigma(u_1) = 20$, and all five edges of C incident to u_1 have weight 4. Let $v \in W$ be another neighbour of u_1. Because $w(u_1v) = 4$ and $\sigma(v) + \gamma(v)$ is odd, we have $\sigma(v)+\gamma(v) \geq 17$. We here decrease $w(u_1v)$ down by 2. Then $\sigma(u_1)$ decreases down to 18, which is still even and now different from $\sigma(v^*) + \gamma(v^*) = 20$. On the other hand, $\sigma(v) + \gamma(v)$ remains odd, and of value at least 15.

- If $d_C(v^*) = 2$, then $\gamma(v^*) = 4 \cdot 3 = 12$, and, since $w(v^*u_1), w(v^*u_2) \in \{3, 4\}$ and $\sigma(v^*) + \gamma(v^*)$ is even, we have $\sigma(v^*) + \gamma(v^*) \geq 18$.

  If $\sigma(v^*)+\gamma(v^*) = 20$, then, when decreasing by 2 one or two of $w(v^*u_1)$ and $w(v^*u_2)$, note that the sum incident to v^* remains at least 16. If both $\sigma(u_1)$ and $\sigma(u_2)$ are equal to 20, then we decrease both $w(v^*u_1)$ and $w(v^*u_2)$ down by 2 so that $\sigma(v^*) + \gamma(v^*)$ gets equal to 16 while both $\sigma(u_1)$ and $\sigma(u_2)$ remain even and equal to 18. Now, if, say, $\sigma(u_1) = 20$ while $\sigma(u_2) \leq 18$ then we decrease $w(v^*u_2)$ down by 2 so that $\sigma(v^*) + \gamma(v^*) = 18$, $\sigma(u_1) = 20$ and $\sigma(u_2) \leq 16$.

  When $\sigma(v^*)+\gamma(v^*) = 18$, we are allowed to decrease only one of $w(v^*u_1)$ and $w(v^*u_2)$ (as otherwise $\sigma(v^*) + \gamma(v^*)$ would become strictly smaller than 15). If only one of $\sigma(u_1)$ and $\sigma(u_2)$ is equal to 18, then the same strategy as in the previous case applies. So assume $\sigma(u_1) = \sigma(u_2) = 18$. Since all edges of C have been weighted with weights among $\{3, 4\}$, this means there is a vertex $v \in W$ different from v^* that is adjacent to u_1, and such that $w(u_1v) = 4$. Since $\sigma(v) + \gamma(v)$ is odd and $w(u_1v) = 4$, we have $\sigma(v) \geq 17$. So we can freely decrease $w(u_1v)$ by 2. When doing so, we still have that

$\sigma(v) \geq 15$ is odd, while $\sigma(u_1)$ is now different from $\sigma(v^*) + \gamma(v^*)$. So we are back in the previous case, and the same arguments apply.

- If $d_C(v^*) = 3$, then $\gamma(v^*) = 4 \cdot 2 = 8$, and, since $w(v^*u_1), w(v^*u_2), w(v^*u_3) \in \{3, 4\}$ and $\sigma(v^*) + \gamma(v^*)$ is even, we again have $\sigma(v^*) + \gamma(v^*) \geq 18$.

  We use arguments that are quite similar to those used in the previous case. The important thing to notice is that if $\sigma(u_i) = \sigma(v^*) + \gamma(v^*)$, then, in C, there are at least two edges, different from u_iv^*, incident to u_i being weighted 4 by w. The second ends of these edges have odd incident sum at least 17, so we can decrease by 2 the weight of one of those edges to decrease $\sigma(u_i)$ by 2.

  If only one of the $\sigma(u_i)$'s, say $\sigma(u_1)$, is equal to $\sigma(v^*) + \gamma(v^*)$, then we consider an edge u_1v with $v \neq v^*$ weighted 4 by w, and decrease $w(u_1v)$ by 2 to get rid of the sum conflict between v^* and u_1. If two of the $\sigma(u_i)$'s, say $\sigma(u_1)$ and $\sigma(u_2)$, are equal to $\sigma(v^*) + \gamma(v^*)$, then we decrease $w(u_3v^*)$ by 2 so that $\sigma(v^*) + \gamma(v^*)$ gets different from $\sigma(u_1)$ and $\sigma(u_2)$, and remains different from $\sigma(u_3)$. Finally, if $\sigma(u_1) = \sigma(u_2) = \sigma(u_3) = \sigma(v^*) + \gamma(v^*)$, then we again consider an edge u_1v with $v \neq v^*$ weighted 4 by w, and decrease $w(u_1v)$ by 2 so that the previous case applies.

- If $d_C(v^*) = 4$, then $\gamma(v^*) = 4$, and, since $w(v^*u_1), w(v^*u_2), w(v^*u_3), w(v^*u_4) \in \{3, 4\}$ and $\sigma(v^*)+\gamma(v^*)$ is even, we have $\sigma(v^*)+\gamma(v^*) \geq 16$. This time, it might be that we cannot decrease by 2 any of the weights of the v^*u_i's so that $\sigma(v^*) + \gamma(v^*)$ remains of value at least 15. However, for every u_i in conflict with v^* there is at least one incident edge u_iv different from u_iv^* being weighted 4, and $\sigma(v) \geq 17$ is odd.

  When $\sigma(v^*)+\gamma(v^*) = 16$, note that by decreasing by 2 any of the $w(v^*u_i)$'s, the sum $\sigma(v^*)+\gamma(v^*)$ gets strictly smaller than 15. In this situation, we do allow $\sigma(v^*)+\gamma(v^*)$ to be smaller than 15, provided it is different from the only value $s \in L(v)$ and from the $\sigma(v^*u_i)$'s. In the case distinction below, we denote by α the initial value of $\sigma(v^*) + \gamma(v^*)$.

  - If only $\sigma(u_1)$ is equal to α, then we decrease by 2 the weight of an edge weighted 4 different from u_1v^* incident to u_1, so that $\sigma(u_1)$ gets different from α, and the other incident sums remain unchanged.
  - If only $\sigma(u_1), \sigma(u_2)$ are equal to α, then we distinguish two cases. If $s \neq \alpha - 2$, then, if possible, we decrease one of $w(v^*u_3), w(v^*u_4)$ by 2 so that $\sigma(v^*)+\gamma(v^*)$ gets equal to $\alpha - 2$ and both of $\sigma(v^*u_3), \sigma(v^*u_4)$ are different from $\alpha - 2$. If this is not possible, it means that $\sigma(v^*u_3) = \sigma(v^*u_4) = \alpha - 2$. In that case, we decrease by 2 three or four of $w(v^*u_1), w(v^*u_2), w(v^*u_3), w(v^*u_4)$ including $w(v^*u_3), w(v^*u_4)$, so that $\sigma(v^*) + \gamma(v^*) \neq s$. Note that no sum conflict may involve v^* and the u_i's, since we get $\sigma(v^*) + \gamma(v^*) \leq \alpha - 6$, and $\sigma(u_1), \sigma(u_2) \geq \alpha - 2$ and $\sigma(u_3), \sigma(u_4) = \alpha - 4$.

    Now assume $s = \alpha-2$. If we are not done when decreasing both $w(v^*u_1), w(v^*u_2)$ by 2, it means that, say, $\sigma(u_3) = \alpha-4$. Now, if we are not done when decreasing both $w(v^*u_1), w(v^*u_3)$ down by 2, then $\sigma(u_4) = \alpha - 4$. Then we are done when decreasing both $w(v^*u_3), w(v^*u_4)$ down by 2. In the latter case, we eventually get $\sigma(u_1) = \sigma(u_2) = \alpha$, and $\sigma(u_3) = \sigma(u_4) = \alpha - 6$, while $\sigma(v^*) = \alpha - 4$ and $s = \alpha - 2$.
  - If only $\sigma(u_1), \sigma(u_2), \sigma(u_3)$ are equal to α, then we reduce down by 2 two or three of $w(v^*u_1), w(v^*u_2), w(v^*u_3)$ so that $\sigma(v^*) + \gamma(v^*) \neq s$. Note that no conflict involving v^* and u_1, u_2, u_3 can arise, since we get $\sigma(v^*) + \gamma(v^*) \leq \alpha - 4$

and $\sigma(u_1), \sigma(u_2), \sigma(u_3) \geq \alpha - 2$. If the resulting $\sigma(v^*) + \gamma(v^*)$ is also different from $\sigma(u_4)$, we are done. Otherwise, we reduce down by 2 the same number of $w(v^*u_i)$'s, including $w(v^*u_4)$, so that we also get $\sigma(v^*) + \gamma(v^*) \neq \sigma(u_4)$.

- If $\sigma(u_1) = \sigma(u_2) = \sigma(u_3) = \sigma(u_4) = \alpha$, then we reduce down by 2 two or three of $w(v^*u_1), w(v^*u_2), w(v^*u_3)$ so that $\sigma(v^*) + \gamma(v^*) \neq s$. Note that $\sigma(u_1), \sigma(u_2), \sigma(u_3)$ remain of value at least $\alpha - 2$, we still have $\sigma(u_4) = \alpha$, while $\sigma(v^*) + \gamma(v^*) \neq s$ and we get $\sigma(v^*) + \gamma(v^*) \leq \alpha - 4$.

In each of these cases, it can be checked that w meets all required properties. □

We finish off the proof of Theorem 1.3. According to Lemma 3.2, we can weight all edges inside/incident to the good H-components in a sum-colouring way, yielding a partial 4-edge-weighting w of G. As pointed out, this edge-weighting w is already sum-colouring from the point of view of the type-1 components of G. Then, for every external vertex v of G, we define its bias $\gamma(v)$ as the current value of $\sigma(v)$, and its list $L(v)$ of forbidden values as $\cup_{u \in H_g \cap N(u)} \{\sigma(u)\}$. In particular, $\gamma(v)$ is a sum of 3's and 4's, since w assigns weights among $\{3, 4\}$ to private edges. Then, according to Lemma 3.3, we can extend w to the edges of the type-2 and type-3 components of G, in such a way that it is sum-colouring from the point of view of these components. Since the resulting w verifies that $\sigma(v) \notin L(v)$ for every external vertex v, we get that w is a sum-colouring 4-edge-weighting of G.

4. Concluding remarks

In this paper, we have provided a weighting strategy for 4-edge-weighting 5-regular graphs in a sum-colouring way. Although this does not prove the 1-2-3 Conjecture for this class of graphs, we believe the method we have used remains of interest, as it stands as a new way of adapting Kalkowski's algorithm to the edge-weighting context.

The most important argument behind the correctness of our weighting scheme is that all vertices of the dominating independent set W are of largest degree in G. This is the key argument which, provided we use the set of weights correctly, guarantees that we can edge-weight the good H-components and the type-2 and type-3 components somewhat independently. It is actually easy to check that, more generally, our weighting scheme can be applied in every graph G with $\Delta(G) = 5$ that admits a dominating independent set whose all vertices have degree 5.

We believe our proof scheme could be adapted to regular graphs with slightly greater degree. Unfortunately, it does not seem obvious to us how to generalize our weighting scheme to general regular graphs. It seems possible to generalize Lemma 3.2 to graphs with maximum degree d, i.e. to show that we can 4-edge-weight G_C in such a way that the incident sums in C are smaller than $3d$. However, we believe Lemma 3.3, for which we did not manage to come up with an easy proof, is hardly generalizable, and it is likely that another type of result is needed here.

References

[1] L. Addario-Berry, R.E.L. Aldred, K. Dalal, B.A. Reed. Vertex colouring edge partitions. *Journal of Combinatorial Theory, Series B*: 94(2):237-244, 2005.

[2] O. Baudon, J. Bensmail, J. Przybyło, M. Woźniak. On decomposing regular graphs into locally irregular subgraphs. *European Journal of Combinatorics*, 49:90-104, 2015.

[3] Y. Duan, H. Lu, Q. Yu. L-Factors and Adjacent Vertex-Distinguishing Edge-Weighting. *East Asian Journal on Applied Mathematics*, 2(2):83-93, 2012.

[4] A. Dudek, D. Wajc. On the complexity of vertex-coloring edge-weightings. *Discrete Mathematics Theoretical Computer Science*, 13(3):45-50, 2011.

[5] Y. Gao, G. Wang, J. Wu. A Relaxed Case on 1-2-3 Conjecture. *Graphs and Combinatorics*, 32(4):1415-1421, 2016.

[6] M. Kalkowski. A note on 1,2-Conjecture. In Ph.D. thesis, 2009.

[7] M. Kalkowski, M. Karoński, F. Pfender. Vertex-coloring edge-weightings: towards the 1-2-3 Conjecture. *Journal of Combinatorial Theory, Series B*, 100:347-349, 2010.

[8] M. Karoński, T. Łuczak, A. Thomason. Edge weights and vertex colours. *Journal of Combinatorial Theory, Series B*, 91:151–157, 2004.

[9] J. Przybyło, M. Woźniak. On a 1,2 Conjecture. *Discrete Mathematics and Theoretical Computer Science*, 12(1):101-108, 2010.

[10] B. Seamone. The 1-2-3 Conjecture and related problems: a survey. Technical report, available online at `http://arxiv.org/abs/1211.5122`, 2012.

[11] C. Thomassen, Y. Wu, C.-Q. Zhang. The 3-flow conjecture, factors modulo k, and the 1-2-3-conjecture. *Journal of Combinatorial Theory, Series B*, 121:308-325, 2016.