Edge-labellings, vertex-colourings and combinatorial games on graphs

Foivos-Sotirios Fioravantes

UNIVERSITÉ CÔTE D'AZUR

ÉCOLE DOCTORALE

SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DE LA COMMUNICATION

THÈSE DE DOCTORAT

Étiquetages d'arêtes, colorations de sommets et jeux combinatoires sur des graphes

Foivos-Sotirios FIORAVANTES

Université Côte d'Azur, CNRS, Inria, I3S, France

Présentée en vue de l'obtention du grade de docteur en Informatique **d'**Université Côte d'Azur

Dirigée par : Nicolas NISSE, Inria Research Officer, Inria Sophia Antipolis
Co-encadrée par : Julien BENSMAIL, Associate professor, Université Côte d'Azur
Soutenue le : 26-09-2022

Devant le jury, composé de :
Olivier TOGNI, Professor, Burgundy University
Éric SOPENA, Professor, LaBRI, Université de Bordeaux
Mariusz WOŹNIAK, Professor, AGH University of Science and Technology
Julien BENSMAIL, Associate professor, I3S, Université Côte d'Azur
Nicolas NISSE, Inria Research Officer, Inria Sophia Antipolis
Théo PIERRON, Associate professor, LIRIS, Université Claude Bernard Lyon 1
Olivier TOGNI, Professor, LIB, Université de Bourgogne

ÉTIQUETAGES D'ARÊTES, COLORATIONS DE SOMMETS ET JEUX COMBINATOIRES SUR DES GRAPHES

Edge-labellings, vertex-colourings and combinatorial games on graphs

Foivos-Sotirios FIORAVANTES

⋈

Jury :

Président du jury
Olivier TOGNI, Professor, Burgundy University

Rapporteurs
Éric SOPENA, Professor, LaBRI, Université de Bordeaux
Mariusz WOŹNIAK, Professor, AGH University of Science and Technology

Examinateurs
Julien BENSMAIL, Associate professor, I3S, Université Côte d'Azur
Nicolas NISSE, Inria Research Officer, Inria Sophia Antipolis
Théo PIERRON, Associate professor, LIRIS, Université Claude Bernard Lyon 1
Olivier TOGNI, Professor, LIB, Université de Bourgogne

Directeur de thèse
Nicolas NISSE, Inria Research Officer, Inria Sophia Antipolis

Co-encadrant de thèse
Julien BENSMAIL, Associate professor, Université Côte d'Azur

Université Côte d'Azur

Foivos-Sotirios FIORAVANTES

Étiquetages d'arêtes, colorations de sommets et jeux combinatoires sur des graphes

xi+202 p.

Étiquetages d'arêtes, colorations de sommets et jeux combinatoires sur des graphes

Résumé

Cette thèse considère deux familles de problèmes définis sur des graphes : les étiquetages d'arêtes propres et les jeux combinatoires. Nous traitons ces problèmes de façon similaire (et classique) : nous montrons que les problèmes considérés sont difficiles à résoudre, puis nous trouvons des algorithmes efficaces sur des instances restreintes.
Nous nous concentrons d'abord sur des problèmes concernant des étiquetages propres de graphes. Pour un entier k fixé, un *k-étiquetage* d'un graphe G est une fonction associant à chaque arête de G une étiquette parmi $\{1, \ldots, k\}$. Un k-étiquetage induit une coloration des sommets de G, où chaque sommet reçoit comme couleur la somme des étiquettes de ses arêtes incidentes. Un k-étiquetage est *propre* si, dans la coloration induite, deux sommets adjacents de G reçoivent des couleurs différentes. D'après la Conjecture 1-2-3, tout graphe connexe d'ordre au moins 3 admet un 3-étiquetage propre. Nous considérons trois variantes de cette conjecture. Nous étudions les k-étiquetages propres équilibrés, pour lesquels les étiquettes assignées apparaissent dans les mêmes proportions. La deuxième variante concerne les étiquetages propres qui minimisent la somme des étiquettes utilisées. Enfin, nous nous intéressons aux 3-étiquetages propres qui minimisent le nombre de fois où l'étiquette 3 est attribuée. Le choix d'étudier ces variantes est naturel. En effet, une version équilibrée de la Conjecture 1-2-3 est que presque tous les graphes G admettent un 3-étiquetage propre équilibré. En outre, la somme des étiquettes d'un tel étiquetage est au plus égale à $2|E(G)|$ et associe l'étiquette 3 à au plus un tiers des arêtes de G. Nous prouvons que les problèmes d'optimisation introduits sont NP-difficiles. Grâce à des résultats structurels et algorithmiques, nous sommes amenés à proposer de nouvelles conjectures pour ces problèmes, que nous vérifions sur quelques classes de graphes (complets, bipartis, réguliers, 3-chromatiques, etc.). Notre travail renforce l'idée que des variantes plus fortes de la Conjecture 1-2-3 pourraient être vraies. Nous terminons en considérant le problème consistant à trouver un plus grand sous-graphe induit d'un graphe donné qui admet un 1-étiquetage propre. Il est prouvé que ce problème est difficile à résoudre et qu'il n'est pas approximable à un facteur $\mathcal{O}(|V(G)|^{1-\frac{1}{c}})$ près pour tout entier c. Néanmoins, nous fournissons des algorithmes paramétrés efficaces.
La deuxième partie de la thèse introduit le *jeu du plus grand sous-graphe connexe Maker-Breaker*, joué par deux joueurs, Alice et Bob, sur un graphe G, initialement non coloré. Les joueurs colorent à tour de rôle les sommets de G, chacun avec sa couleur, jusqu'à ce que tous les sommets soient colorés. Alice est la gagnante si, à la fin, le plus grand sous-graphe connexe de G induit par sa couleur est d'ordre au moins k, un entier fixé. Sinon, Bob gagne le jeu. Nous considérons aussi une version Score du même jeu, dans laquelle le gagnant est le joueur dont la couleur induit le plus grand sous-graphe connexe de G à la fin du jeu. Nous prouvons que décider de l'issue de ces deux jeux est PSPACE-difficile et nous fournissons des algorithmes efficaces pour le cas où le jeu se déroule dans certaines familles de graphes (chemins, cycles, cographes, $(q, q-4)$-graphes, etc.). En comparant ces deux jeux, la principale différence que nous observons est que Bob ne peut jamais gagner la version Score (si Alice joue de manière optimale). Pour une valeur de k égale à la moitié de l'ordre de G, remarquons que si Alice peut gagner la version Maker-Breaker alors elle peut aussi construire un sous-graphe connexe du même ordre dans la version Score ; de tels graphes sont nommés A-parfaits. Nous étudions les graphes réguliers qui sont A-parfaits et prouvons que tout graphe 3-régulier A-parfait a au plus 16 sommets. Nous terminons en fournissant des conditions suffisantes pour qu'un graphe soit A-parfait.

Mots-clés : Graphe, Coloration, Conjecture 1-2-3, Jeux combinatoires, Complexité.

Edge-labellings, vertex-colourings and combinatorial games on graphs

Abstract

In this thesis, we consider two families of computational problems defined on graphs: proper edge-labellings and combinatorial games. We attack these problems in a similar (and classical) way: we show that they are computationally hard, and then find efficient algorithms for instances with specific structure.
First we focus on problems related to proper labellings of graphs. For some natural number k, a *k-labelling* is a weight function on the edges of a graph G, assigning weights, called labels in this context, from $\{1, \ldots, k\}$. A k-labelling induces a vertex-colouring of G, where each vertex receives as colour the sum of the labels of its incident edges. A k-labelling is *proper* if the induced vertex-colouring is proper, *i.e.*, such that any two adjacent vertices of G are assigned different colours. According to the so-called 1-2-3 Conjecture, any connected graph of order at least 3 should admit a proper 3-labelling. We consider three variations of this conjecture. We look into equitable proper k-labellings, for which the assigned labels appear an equal number of times. We then focus on proper labellings that also minimise the sum of labels being used, and finally, proper 3-labellings that also minimise the number of times that the label 3 is assigned. The choice to study these variations is natural. Indeed, an equitable version of the 1-2-3 Conjecture claims that almost every graph G should admit an equitable proper 3-labelling. Also, the sum of labels of such a labelling would be at most $2|E(G)|$ and it would assign label 3 to at most one third of the edges of G. We prove that the introduced optimisation problems are NP-hard. Furthermore, through structural and algorithmical results, we propose new conjectures for the upper bounds of the parameters that we study, which we verify for specific graph classes (*e.g.* complete, bipartite, regular, 3-chromatic, etc.). Interestingly, our work gives further evidence that stronger variations of the 1-2-3 Conjecture could hold. We close our study of proper labellings by considering the problem of finding a largest induced subgraph of a given graph that admits a proper 1-labelling. This problem is proven to be computationally hard and not approximable within a ratio of $\mathcal{O}(|V(G)|^{1-\frac{1}{c}})$ for every natural number c. Nevertheless, we provide efficient parameterised algorithms.
In the second part of the thesis, we introduce and study the *Maker-Breaker largest connected subgraph game*. This game is played by two players, Alice and Bob, on a shared, initially uncoloured graph G. The two players take turns colouring the vertices of G, each one with their own colour, until there remains no uncoloured vertex. Alice is the winner of the game if, by the end, the largest connected subgraph of G induced by her colour is of order greater than k, where the natural number k is also given at the start of the game. Otherwise Bob wins the game. We also consider a Scoring version of the same game, played in the same way, but in which the winner is the player whose colour induces the largest connected subgraph of G by the end of the game. We first prove that deciding the outcome of both of these games is PSPACE-hard, and then proceed by providing efficient algorithms when the games are played on particular graph classes (*e.g.* paths, cycles, cographs, $(q, q-4)$-graphs, etc.). Comparing the behaviour of these games, one of the main differences we observe is that Bob can never win the Scoring version (if Alice plays optimally). Nevertheless, if Alice can win the Maker-Breaker version when playing on G for a value of k equal to half the order of G (the best outcome she can hope for), then she can build a connected subgraph of the same order for the Scoring version; such graphs are called A-perfect. We then study regular graphs that are A-perfect and prove that any 3-regular A-perfect graph has order at most 16. We finish by providing sufficient conditions for a graph to be A-perfect.

Keywords: Graph, Colouring, 1-2-3 Conjecture, Combinatorial games, Complexity.

Acknowledgements

Here is the part where I will *try*, and most probably fail, to thank everyone who has played a significant role in my life up to this point. I stress the word *try*, because I consider this to be an almost impossible task.

So I will start with the easy part, which is, of course, to thank my two supervisors Julien Bensmail and Nicolas Nisse, or Nicolas Nisse and Julien Bensmail (I cannot decide who to name first, and don't want to go by alphabetical order). Both of you have helped me immensely during the last 3.5 years, something for which a will always be grateful, especially considering how particular most of these past years have been (I refrain from sullying my manuscript with the cause of the particularity of the last years). Both of you have taught me many many many things (just one many was indeed not enough here), a good part of which does not relate to computer science or mathematics.

The next easy part is of course to thank all the members of my jury: Théo Pierron, Olivier Togni, and, especially, Éric Sopena and Mariusz Woźniak who did me the immense honor of reviewing this document and offered some most valuable feedback on my work. Thank you all for being interested by my work and accepting to take part in what will surely be a very interesting experience (the defense).

Another easy part: the people with which I had the joy of sharing my office these past years: Thibaud and Lucas. There are many happy moments that come to mind with each of you two, from the "souffleur" that was not working when Thibaud was present, to Lucas suffering from Tarsi's Conjecture.

And now starts the hard par: all the remaining COATIs. Please do not be mad at me if I don't mention your name. If I mentioned all the good memories I have with most of you, the amazon would not have enough trees remaining after I printed this manuscript. I will nevertheless mention at least Thomas and the rabbits, Francesco and walking and pizzas, Igor and the great nights out, Michel and the 6 cylinders, Christelle and the Foivó, Majus and the Brooks Theorem, Joanna and the "let's take the long way to come down from the mountain", David and driving to and from la Rochelle (with some great music), Aurora and the tiramisu, and the list goes on and on. And of course Fionn and Brussels (also for the fun times we spent working together). I will also have to reserve a special place for Jean-Claude Bermond, without which I wouldn't even have found myself in Nice, let alone get to know all of you.

Last (for the english part), but not least, most (again many will unavoidably be missing here) of the people who, in some sense, have also contributed in me being who I am today: Vanjul, Michelangelo, Guilherme, Sara, Antonia, Andrea, Alessio, Piergiorgio, Washy, and again the list goes on and on.

Και τώρα ήρθε η ώρα για λάθη ορθογραφίας. Π*ρώτα πρώτα θα ευχαριστήσω τους γονείς μου,* Σ*πυροβασίλειο και Ελένη, καθώς και την θεία μου,* Π*αρασκευή. Ό, τι και να σχολιάσω, ό, τι και να ευχαριστήσω, θα είναι πολύ λίγο.* Ύ*στερα θα αναφερθώ στο τρίο στούτζες, τα άλλα δυο μέλη του οποίου είναι ο* Θ*εόφιλος και ο Νίκος.* Π*ροφανώς θα ευχαριστήσω και τους Άρη* Π*αγουρτζή και* Σ*τάθη Ζάχο , αλλά και τον μεγάλο* Δ*άσκαλο Αλέξανδρο* Π*απαιωάννου, χωρίς τους οποίους δεν θα είχα αγαπήσει*

ίσως ποτέ τα διακριτά μαθηματικά/θεωρητική πληροφορική και, πιο συγκεκριμένα, την Θεωρία γραφημάτων. Μετά θα αναφερθώ στην ομάδα “απογεύματα Κυριακής”: Δημήτρης, Σωτήρης, Καθολική, Θεόφιλος και Γιάννης, και τις πολλές και ιδιαίτερα δημιουργικές ώρες που περάσαμε μαζί τα τελευταία χρόνια. Και τέλος ήρθε η στιγμή που θα αραδιάσω ονόματα, χωρίς σε καμία περίπτωση αυτό να σημαίνει ότι οι ακόλουθοι δεν έχουν επηρεάσει, ο καθένας με τον τρόπο του, καθοριστικά την ζωή μου : Ελεονώρα, Μαρία, Πέτρος, Γιάννα, Κώστας, Παρασκευή, Τάσος, Βέρα, Κωστής, Αλεξάνδρα, Γεωργία, Αντώνης, Γιάννης κτλ.

Σας ευχαριστώ όλους. I thank each and every one of you. Un grand merci à vous tous.

Contents

CHAPTER 1

Introduction

All these difficulties are but consequences of our refusal to see that mathematics cannot be defined without acknowledging its most obvious feature: namely, that it is interesting.

— Michael Polanyi, [102] (p.188).

This thesis treats problems arising in *graph theory*. In particular, we consider two families of combinatorial problems defined on graphs. First, we consider *proper labellings* of graphs, a notion which falls under the general domain of *graph colouring*. Then, we introduce and study a new combinatorial game in which the players strive to build a largest connected subgraph of a given graph.

1.1 Colouring and playing combinatorial games on graphs

The first part of this thesis focuses on finding proper labellings of graphs. On a high level description, such labellings are weight functions on the edges of a given graph G, which can be used in order to distinguish the adjacent vertices of G. In other words, these labellings can help us to put colours on the vertices of G, so that no two adjacent vertices of G have the same colour. Constructing such colourings falls in a classical domain of graph theory, that of *graph colouring*.

In the setting of graph colouring, the usual task is to construct *colouring functions*, which assign natural numbers, usually referred to as *colours*, on some elements of a given graph, such that the colours verify some desired property. For example, given a graph $G = (V, E)$, one problem could be to find a colouring $c : V \to \{1, \dots, k\}$ (for any $k \in \mathbb{N}$) such that for every edge $v_1v_2 \in E$, $c(v_1) \neq c(v_2)$. Such a colouring is usually referred to as a *proper k-vertex-colouring* of G. It is fairly easy to see that one can always find a proper k-vertex-colouring of a graph G, for some $k \in \mathbb{N}$, as it suffices to have c assign a unique colour to each vertex of G. Things however become way more interesting when the task is to, given a graph G, find the *minimum* k such that G admits a proper k-vertex-colouring, usually called the *chromatic number* of G and denoted as $\chi(G)$.

The problem of calculating the chromatic number of a graph G is among the most fundamental problems of computer science. The decision version of this problem, *i.e.*, given a graph G and a natural number $k \geq 3$, decide if $\chi(G) \leq k$, is NP-complete, even if G is planar [74], while it is polynomial for the case $k = 2$, properly 2-colourable graphs being all bipartite, a property that is easy to detect. Moreover, this is one of the most well-known problems of graph theory, mainly due to the so-called *four colour theorem*, stating that any planar graph has chromatic number at most 4.

The four colour theorem was initially proposed as a conjecture by Francis Guthrie in 1852. After an initial failed proof by Alfred Kempe (which was not completely without importance as it inspired Heawood to prove the “five colour theorem” in [76]), the theorem was finally proven by Appel and Haken in [8, 9]. However, their proof received a lot of criticism from the scientific community, as it is one of the first examples of a computer-aided proof. It is worth noting that, although improved versions of the proof by Appel and Haken have been proposed, *e.g.* in [110], up to this day there exists no proof of the four colour theorem which does not require the use of a computer.

Graphs G that verify $\chi(G) \leq 3$ are of great importance for some of the results presented in this thesis (see Section 2.2.1 for more details). However, deciding if a graph verifies this property is infeasible in general (unless P=NP). It is thus important to identify classes of graphs that do indeed verify this property. One such family, according to Grötzsch’s Theorem [75], is that of triangle-free planar graphs.

Up to this point, the focus was on constructing colouring functions that distinguish the adjacent vertices of a given graph. But it is worth noting that this is not the only interesting problem that can be defined in the field of graph colouring. Another example of a problem in this setting could be to, given a graph $G = (V, E)$, find a colouring $c : E \rightarrow \{1, \dots, k\}$ (for any $k \in \mathbb{N}$) such that for every pair of edges $e_1, e_2 \in E$, if e_1 and e_2 share a vertex in V then $c(e_1) \neq c(e_2)$. Such a colouring is usually referred to as a *proper k-edge-colouring* of G. The minimum k such that G admits a proper k-edge-colouring is called the *chromatic index* of G, and usually denoted as $\chi'(G)$.

The problem of finding an optimal, *i.e.*, which utilises the minimum number of distinct colours, proper edge-colouring of a graph, behaves in a similar way as that of finding an optimal proper vertex-colouring. Nevertheless, proper edge-colourings are somewhat better understood than proper vertex-colourings, as can be attested by the Vizing’s Theorem [117], stating that for every graph G with maximum degree Δ, $\Delta \leq \chi'(G) \leq \Delta + 1$. Note that this theorem does not provide a complete characterisation of which graphs have chromatic index equal to Δ or $\Delta + 1$, but almost all graphs belong in the first case [66].

We are now ready to return to the notion of proper labellings of graphs. In a nutshell, a k-labelling of a graph G is a function that takes numbers, called labels in this context, from $\{1, \dots, k\}$ and assigns them on the edges of G. A k-labelling is said to be *proper* if the vertex-colouring where each vertex receives as a colour the sum of the labels of its incident edges, is itself a proper vertex-colouring of G. By $\chi_\Sigma(G)$ we denote the minimum k so that there exists a proper k-labelling of G. In some sense, a proper labelling can be viewed as a (not necessarily proper) edge-colouring from which a proper vertex-colouring is produced. We stress at this point that a proper labelling of a graph G is not necessarily a proper edge colouring of G.

The first major family of problems studied in this thesis revolves around the χ_Σ parameter. The authors of [81] propose an intriguing conjecture, according to which any *nice* graph G, *i.e.*, a connected graph of order at least 3, should verify $\chi_\Sigma(G) \leq 3$. Informally, this conjecture could be interpreted as: “it should always be possible to find a proper vertex-colouring of any nice graph, only by combining the natural numbers $1, 2$ and 3 in a smart way”. The interested reader is invited to see Section 1.2 for a formal definition of this conjecture, which is known as the *1-2-3 Conjecture*. Already, the fact that $\chi_\Sigma(G)$ is claimed to be upperly bounded by a constant number, let alone a number equal to 3 (in complete contrast to $\chi(G)$ which according to Brooks’ Theorem [40] can be of the same order as the maximum degree of the graph), is quite astounding.

Nevertheless, the 1-2-3 Conjecture seems rather plausible, with $\chi_\Sigma(G) \leq 5$ for every nice graph G having been proven in [79] *.

Chapters 3, 4 and 5 of this thesis deal with three more restricted variations of the χ_Σ parameter. Apart from the individual interest that the study of these variations has, they are also interconnected, meaning that each one of these variations leads naturally to the definition and study of the next one. More importantly, our work on these variations supports the idea that, even if the 1-2-3 Conjecture were to eventually be proven correct, depending on how this was supposedly achieved, we could still be far from fully grasping the notion of proper labellings of graphs. These variations, as well as their interplay, are formally presented in Chapter 2, along with some important techniques that we will employ throughout the first part of this thesis. The first part of the thesis closes with Chapter 6, which is about finding a largest induced subgraph G' of a given graph G, so that $\chi_\Sigma(G') = 1$.

The second part of this thesis introduces and studies two new combinatorial games on graphs. As concrete examples of such games, consider the following two games that deal with graph colouring, introduced in [37]. In both two-player games, both players take turns colouring the vertices of a common graph in a prespecified order, using colours from a predefined set of colours and making sure that the newly coloured vertex does not share its colour with any of its neighbours. The difference between these two games lies in the goal of the players: for the first game, the first player that is unable to colour a vertex loses the game, while for the second game, the first player wins if and only if all the vertices of the graph are coloured by the end of the game. The second one of these games is particularly interesting as, if we drop the requirement on the players to colour vertices in a pre-specified order, it studies the scenario where we try to construct a proper vertex-colouring of a graph, all the while somebody else is maliciously colouring vertices in order to stop us. A first version of this game, known as the *Colouring construction game*, had already been introduced in [111].

The Colouring construction game can be used to study the problem of constructing a proper vertex-colouring of a given graph G, when roughly half of the colours are chosen in an arbitrary way. The interesting parameter which is considered here is the minimum number of colours such that the first player can always win this game when played on a given graph G, named the *game-chromatic number* of G. It is proven in [37] that the game-chromatic number of any tree is at most 5, while it is left as a conjecture that there exists a constant c such that any planar graph has game-chromatic number at most c, which would be the combinatorial game theoretic equivalent to the four colour theorem. This conjecture was proven to be correct in [84] for $c = 33$, which was later improved, with the best known upper bound being $c = 17$ [123], while we know that there exist planar graphs in which the first player requires 11 colours in order to win [122].

Games like the Colouring construction game belong to the family of *two-person perfect-information games* ,*i.e.*, games played by two players on the same structure (usually represented by a graph), where both players have full knowledge of their adversary's moves, and no move is done in a random way (or depends on luck). Such games are studied in the domain of *combinatorial game theory*.

Another very well-known example of a combinatorial game, is the game of tic-tac-toe, which is rather beloved by young students around the world. However, after playing for some time, most

*. Recently, a new work has appeared on arXiv, in which it is claimed that $\chi_\Sigma(G) \leq 4$ is proven for every nice graph G [82]. As far as the author can tell, the result seems correct. Keep in mind, however, that this work has not yet passed the peer reviewing process.

people lose their interest in playing this game. This is because, for some seemingly inexplicable reason, after having played a decent number of games against their friends, no one seemed to be able to win any more; most of the games of tic-tac-toe seemed to lead to a draw. Actually, this is something that can be mathematically proven, using relatively simple arguments. We refer the reader to [35] for an excellent analysis of this game.

One of the main motivations behind studying such games, is that they are a source of computationally hard problems. Even and Tarjan claim that «This result suggests that the theory of combinatorial games is difficult» [67]. The result they are referring to is that deciding who wins in the *Shannon switching game* (a variation of the well known board game named *Hex*) is PSPACE-complete when played on the vertices of a given graph. Such games are discussed in more length in Section 1.3.1. These are just some of the combinatorial games which are PSPACE-complete, with many more appearing, *e.g.* in [111].

In Chapters 7, 8 and 9, we introduce and study the *largest connected subgraph game*. This game is played on a shared graph G, in which both players strive to create the largest connected subgraph of G. We also introduce a variation of this game, in which the goal of the first player is the same as above, but the second player tries their best to stop the first player from achieving their goal. Both variations of this game are natural problems to propose which, surprisingly, no one had done up to this point.

We proceed now to formally introduce the notions treated in this thesis.

1.2 1-2-3 Conjecture

We start by properly introducing the notations and the notions concerning the χ_Σ parameter. For notions and definitions on graph theory not explained here, we refer the reader to [59]. From now on, whenever we talk about a proper colouring of a graph, we mean a proper vertex-colouring. Also, unless mentioned otherwise, any graph we mention is assumed nice.

Let $G = (V, E)$ be a graph. A function $\ell : E \to \{1, \dots, k\}$ is called a *k-labelling* of G. In this thesis, we are more particularly interested in *proper labellings*, which are defined as follows. For any $v \in V$, denote by $c_\ell(v)$ the *colour* of v that is induced by ℓ, being the sum of labels assigned to the edges incident to v. That is,

$$c_\ell(v) = \sum_{u \in N(v)} \ell(vu),$$

where $N(v) = \{u \in V : uv \in E\}$ is the neighbourhood of v. The labelling ℓ is said to be *proper* if the resulting c_ℓ is a proper vertex-colouring of G, *i.e.*, for every edge $uv \in E$, $c_\ell(u) \neq c_\ell(v)$. Note that a graph admits a proper labelling only if it contains no K_2 as a connected component [81]. Indeed, consider the graph K_2, *i.e.*, a graph with only two vertices u, v connected through a single edge uv. Let ℓ be an arbitrary labelling of K_2. It follows that $c_\ell(u) = c_\ell(v)$, and this is regardless of what label was assigned to the edge uv. Therefore, proper labellings can only be defined on *nice graphs*, *i.e.*, graphs without any connected component isomorphic to K_2. It is clear that for every (not necessarily proper) labelling ℓ of G and for every $v \in V$, $c_\ell(v) \geq d(v)$ with $c_\ell(v) = d(v)$ if and only if all edges incident to v are assigned label 1. It follows that G admits a proper 1-labelling if and only if for every edge $uv \in E(G)$, $d(u) \neq d(v)$. As an example for such a graph, consider P_3, the path on three vertices. Graphs that verify this property are named *locally irregular* and form an important family in this domain (more details on these graphs can be found in Section 1.2.2). The problem treated in Chapter 6 focuses on locally irregular graphs.

The connection between the labellings of a graph G and its "regularity" has been explored by several authors, in particular through the notion of irregularity strength of graphs, introduced by Chartrand et al. in [50]. The irregularity strength of a graph G, denoted as $s(G)$, can be defined as the smallest k such that G admits a k-labelling ℓ with the property that for every two $u, v \in V$ (not necessarily adjacent), $c_\ell(u) \neq c_\ell(v)$. One of the main points for studying such labellings ℓ, is that replacing each edge e of G by $\ell(e)$ parallel edges, results a multigraph M containing G as a subgraph, which has a very interesting property: all the vertices of M have different degrees. Such multigraphs are called *irregular*. It is fairly known that no non-trivial simple graph G is irregular; this can be proven through an application of the pigeonhole principle. It is thus legitimate to wonder how to build a corresponding irregular multigraph M in the above fashion. In their work, Chartrand et al. regard edge multiplications as an expensive operation and, as such, they want to limit it as much as possible. This results in the following optimisation problem: for a given graph G, what is the smallest k such that G can be turned into an irregular multigraph M by replacing each edge with at most k parallel edges? From the labelling point of view, this smallest k is precisely $s(G)$.

Another interesting family of graphs to consider in this context, is that of regular graphs as, in some sense, they lie in the opposite side of irregularity (all their vertices having the same degree). Formally, for $d \geq 1$, a graph G is said to be *d-regular* if every vertex of G has degree exactly d. Indeed, this family of graphs was already considered in [68], where it was shown that any d-regular graph verifies $s(G) \leq \lceil \frac{n}{2} \rceil + 9$, where n is the order of G. In the same work, it was conjectured that there should exist a constant c such that $s(G) \leq \lceil \frac{n}{d} \rceil + c$, for any d-regular graph G. There have been some steps towards this conjecture, with $s(G) \leq 16\frac{n}{d} + 6$ being proven in [103], which was later improved to $s(G) \leq 6\lceil \frac{n}{d} \rceil$ in [80].

Since the notion of irregularity does not fit with simple graphs, as K_1 is the only irregular simple graph, it is legitimate to consider that this notion is too strong for simple graphs, and instead consider weaker notions of irregularity. A few such notions have been explored in the literature, such as the notions of highly irregular graphs [48] or locally irregular graphs [17].

Coming back to proper labellings of graphs, the main parameter that will interest us in the first part of this thesis, is the minimum k so that a given graph G admits a proper k-labelling, denoted as $\chi_\Sigma(G)$. This parameter χ_Σ is precisely at the heart of one of the most famous conjectures concerning proper labellings of graphs, the so-called *1-2-3 Conjecture*, introduced by Karoński, Łuczak and Thomason in 2004 [81]:

1-2-3 Conjecture. *If G is a nice graph, then $\chi_\Sigma(G) \leq 3$.*

In simple words, this conjecture claims that, for every nice graph G, regardless of how large $\chi(G)$ is, it should be possible to assign labels $1, 2$ and 3 to its edges in order to define a proper vertex-colouring of G. There exists a number of results that point towards the correctness of this conjecture.

First, note that there exist nice graphs G that verify $\chi_\Sigma(G) = 3$. For example every nice complete graph K_n has $\chi_\Sigma(K_n) = 3$ as shown in [50]. The same holds true for every odd-length cycle (see upcoming Chapter 4 for a simple proof of this fact). The most famous result that supports the 1-2-3 Conjecture belongs to Kalkowski, Karoński and Pfender [79], and states that for any nice graph G, $\chi_\Sigma(G) \leq 5$ [†]. Another important result is in [105], where it is shown that for every nice regular graph G, $\chi_\Sigma(G) \leq 4$. A classic result in this domain is in the original paper about the 1-2-3 Conjecture [81], where it is shown that for every nice 3-colourable graph, $\chi_\Sigma(G) \leq 3$. The proof of this last result is quite useful, as with just slight modifications of that proof, one can construct new proper labellings of 3-colourable graphs, that verify additional constraints. For that reason, one version of that proof will be presented in Section 2.2.1. However, unless the given nice graph G is locally irregular (in which case $\chi_\Sigma(G) = 1$), deciding whether $\chi_\Sigma(G) \leq 2$ holds is NP-complete [64]. This remains true even if G is assumed to be a cubic graph [58]. Thus, unless P=NP, there is no "good" characterisation of graphs admitting proper 2-labellings (or, the other way around, of graphs needing 3s in their proper 3-labellings). Nevertheless, a good characterisation of nice bipartite graphs G with $\chi_\Sigma(G) = 3$, was provided in [116]. This result, together with the proof that bipartite graphs verify the 1-2-3 Conjecture [81], completes our understanding of the behaviour of these graphs (at least for the original version of the 1-2-3 Conjecture).

We close this section by mentioning some variations of the problem of finding proper k-labellings that we find interesting. We refer the interested reader to [112] for even more variations that are not mentioned here.

We begin by considering the problem of finding proper 2-labellings where we are also allowed to alter the induced colour of each vertex by increasing it by 1 or 2. More precisely, a proper total labelling differs from a proper labelling in that in the former the vertices also receive a label that affects their induced colour. Such labellings are known as *total* proper k-labellings, and their study was initiated in [106]. The main motivation behind the introduction of total labellings is that, in general, using three labels might be too powerful to find a proper labelling, in the sense that two labels are always "almost enough". This is supported by results such as the ones in [2], where it is proven that for any fixed $p \in (0, 1)$, the random graph $G_{n,p}$ (asymptotically) almost surely admits a proper 2-labelling. It is worth mentioning here that the authors of [106] also propose a conjecture concerning proper total labellings:

1-2 Conjecture. *If G is a nice graph, then G admits a proper total 2-labelling.*

To date, no counterexample is known for the 1-2 Conjecture. A break-through result for this conjecture was proven by Kalkowski [78], who showed that every nice graph G admits a total proper 3-labelling. It should be noted that this result is actually what made it possible to prove that $\chi_\Sigma(G) \leq 5$ for every nice graph G [79].

Recall that in a k-labelling of a graph G, all the edges of G are assigned a label from the set $\{1, \dots, k\}$. A different variation of labellings, known as *list* labellings, is proposed in [15], according to which a set $L(e)$ of possible labels is defined for every edge e of G, and a labelling must be constructed so that every edge is assigned a label from its list. If there exists a function ℓ such that for each edge e of the given graph G, $\ell(e) \in L(e)$ and ℓ is also a proper labelling, we say that G admits a proper labelling from L. We will also say that G is *k-labelling-choosable* if G admits a proper labelling from every list L with $|L(e)| \leq k$ for every $e \in E(G)$. The authors of [15] propose the following conjecture:

List 1-2-3 Conjecture. *If G is a nice graph, then G is 3-labelling-choosable.*

†. We refer the interested reader to [21], Section 2.2, for a clear explanation of the mechanisms that lie behind the proof presented in [79].

It is worth noting that although there had been some steps towards the above conjecture [60], it was actually not known until recently whether there even exists a constant c such that any nice graph is c-labelling-choosable. The first such result was proven in [45], and it is for $c = 17$. An improved bound (for $c = 5$) was also proposed in [124]. It is also worth mentioning [121] and [107] where the authors consider total variations of proper list labellings, and that proper total labellings also play an important role in both [45] and [124].

The variations mentioned up to now differ from proper labellings by considering different ways of assigning integers on the vertices/edges of a graph. The last set of variations we would like to mention differ from proper labellings by instead changing the way that the induced colours are produced.

The authors of [1] study the problem of k-labellings such that for each pair of adjacent vertices u and v, the multiset defined by the labels of the edges that are incident to u is different from the one defined by the labels of the edges that are incident to v. Let us say that a k-labelling verifying this property is a proper *multiset* k-labelling. It was shown in [1] that every nice graph admits a proper multiset 4-labelling. This was improved relatively recently in [118], where it was proven that every nice graph admits a multiset proper 3-labelling.

Finally, the author of [115] studied k-labellings ℓ where the induced colour of each vertex is produced by multiplying the labels of its incident edges (instead of adding them). If the resulting colouring is proper, we say that ℓ is a proper *product* k-labelling. Among other results, it was also proposed in [115] that every nice graph should admit a proper product 3-labelling. This conjecture was proved quite recently in [30]. It is worth mentioning that this proof draws inspiration from the one presented in [118] for the multiset version explained above. Also note that both of the results proven in [118] and in [30] are tight as there are graphs, for example the complete graphs, that require at least 3 labels by any proper multiset or product labelling.

1.2.1 Variations with additional optimisation

Recently, there is a new line of research that has started emerging, which is dedicated to studying optimisation problems related to the 1-2-3 Conjecture which arise when investigating the existence of proper labellings fulfilling additional constraints. One of the main sources of motivation here is to further understand the very mechanisms that lie behind proper labellings.

Consider now a graph that is the cycle on four vertices C_4. Clearly, $\chi(C_4) = 2$. Figure 1.1(a) illustrates a proper 2-colouring c of C_4. Also, C_4 is not locally irregular, from which follows that $\chi_\Sigma(C_4) > 1$. Moreover, Figure 1.1(b) provides a proper 2-labelling ℓ of C_4. Thus, $\chi_\Sigma(C_4) = 2$. Observe however that there is an important difference between Figures 1.1 (a) and (b): the maximum colour used by c is equal to 2, while the maximum colour induced by ℓ is equal to 4. Also, c uses only two distinct colours, while ℓ induces three different colours. So the natural question to be asked here is whether these differences stem from the choice of ℓ provided above, or whether it is a more general phenomenon.

In order to better understand the connection between proper labellings and proper vertex-colourings, the authors of [16, 31] studied proper labellings ℓ for which the resulting vertex-colouring c_ℓ is required to be close to an optimal proper vertex-colouring (*i.e.*, with the number of distinct resulting vertex colours being close to the chromatic number). Notably, the authors of [31] prove that for every nice tree T of maximum degree Δ, the minimum maximum colour that is induced by any proper 2-labelling of T is a value in $\{\Delta, \Delta + 1, \Delta + 2\}$, which can be arbitrarily larger than $\chi(T) = 2$.

(a) A proper colouring of C_4, using two colours.

(b) A proper 2-labelling of C_4

Figure 1.1 – A proper 2-colouring c and a proper 2-labelling ℓ of C_4. In subfigure (a), the small numbers in the vertices correspond to the colours assigned by c. In subfigure (b), the small numbers on the edges correspond to the labels assigned by ℓ, while the small numbers in the vertices correspond to the colour of the corresponding vertex induced by ℓ.

Each of these previous investigations led to presumptions of independent interest. In particular, it is believed in [31], that every nice graph G should admit a proper labelling where the maximum vertex colour is at most $2\Delta(G)$, while our own research (see Chapter 4) seems to suggest that every nice graph G should admit a proper labelling where the sum of assigned labels is at most $2|E(G)|$. One of the main reasons why these presumptions are supposed to hold, is the fact that, in general, it seems that nice graphs admit 2-labellings that are almost proper, in the sense that they only need a few 3s to design proper 3-labellings.

It is also worth mentioning that this belief on the number of 3s is actually a long-standing one of the field, as, in a way, it lies behind the 1-2 Conjecture. One way to interpret the 1-2 Conjecture is that for any nice graph, it should be possible to design proper 2-labellings if the colour of every vertex is also allowed to be slightly altered. This local alteration could be achieved by, for example, assigning label 3 to only a few edges of the graph. As a consequence, to the best of our knowledge there is no known (non-trivial) graph G with $\chi_\Sigma(G) = 3$ for which the three labels $1, 2, 3$ must be used with nearly equal proportion in every proper 3-labelling. As an example, let us mention that for every nice complete graph K_n, for which $\chi_\Sigma(K_n) = 3$, there is a proper 3-labelling assigning label 2 only once [31]. Also, for every bipartite graph G with $\chi_\Sigma(G) = 3$, there exist proper 3-labellings assigning label 3 at most twice (Chapter 4).

It is important to mention a take on those questions by Baudon, Pilśniak, Przybyło, Senhaji, Sopena, and Woźniak. In [18], they investigated proper labellings in which all labels must be assigned about the same number of times. A labelling ℓ is called *equitable* if, for every two distinct labels α, β assigned by ℓ, the number of edges assigned label α differs from the number of edges assigned label β by at most 1. The smallest k such that G admits an equitable proper k-labelling is denoted by $\overline{\chi_\Sigma}(G)$. This parameter $\overline{\chi_\Sigma}$ is defined for every nice graph, although this is not trivial to be shown.

The authors of [18] have investigated several aspects of equitable proper labellings, most of which are about the relationship between $\chi_\Sigma(G)$ and $\overline{\chi_\Sigma}(G)$ for a given graph G. For a few families of graphs G, they have notably established that $\overline{\chi_\Sigma}(G) = \chi_\Sigma(G)$ holds, except for a few exceptions. In this thesis, we will mainly focus on investigating the graphs G with $\chi_\Sigma(G) < \overline{\chi_\Sigma}(G)$. Also, it is legitimate to ask whether an equitable version of the 1-2-3 Conjecture is plausible.

1.2.2 Locally irregular graphs

As explained earlier, the class of locally irregular graphs can be seen as an antonym to that of regular graphs. Recall also that there exist several additional such notions. This is mainly due to the fact that non-trivial irregular graphs do not exist. Thus, the literature has a multitude of slightly different definitions of irregularity (see for example [4, 48, 50, 70, 103]). One way to deal with the nonexistence of irregular graphs, is to restrain the definition of irregularity. Intuitively, instead of demanding for all vertices of a graph to have different degrees, each vertex v is now considered separately, and the request is that v and/or the vertices "around" v verify some property of irregularity. For example, the authors of [5] study graphs with the following property: for every vertex v of a graph G, no two neighbours of v have the same degree. For an overview of other interesting notions of irregularity (local or otherwise), we refer the reader to [6]. The main notion which will be treated in this thesis, and is going to be the main focus of Chapter 6, is that of locally irregular graphs.

Recall that a graph G is said to be locally irregular, if every two adjacent vertices of G have different degrees. The notion of locally irregular graphs was first introduced in [17]. The reason why locally irregular graphs are of interest in this thesis comes from their connection to the 1-2-3 Conjecture. An obvious connection is that this conjecture holds for locally irregular graphs. Furthermore, there have been taken some steps towards proving that conjecture, which involve edge-decompositing a graph into a constant number of locally irregular subgraphs, *i.e.*, given G, find an edge-colouring of G using a constant number of colours, such that each colour induces a locally irregular subgraph of G. This is the main motivation behind [17], and it remains interesting enough to attract more attention [33, 92, 104].

In the next section, we introduce the main problems studied in the second part of this thesis.

1.3 The largest connected subgraph games

Chapters 7, 8 and 9 present our work on two variations of a certain 2-player combinatorial game. To define these games, consider the following scenario: two players, Alice and Bob, want to play a famous card game. Now in order to actually play the game, they have to first construct their respective decks. To build their decks, they have to use cards from the *library*, a finite collection of cards which is common for both of them. To pick cards, the players have to follow a *drafting* procedure: they take turns choosing cards from the library to include to their respective decks, with the obvious constraint that once a card is picked by one of the players, the other no longer has the right to pick it. Once there are no more cards remaining in the library, Alice and Bob can go on and play the actual game. It is important to note that not all the cards of the library are "synergistic", meaning that some pairs of cards are more useful than other. So this begs the question: how should Alice or Bob choose their respective cards during the drafting procedure? Should they try to follow some greedy strategy, always trying to maximise the synergy of the cards they are picking? Should they follow a more "aggressive" strategy and try to pick cards that their opponent would like to have? Should they try mixing up these two kinds of strategies?

One way to try to answer the above questions could be through modeling the above situation as a problem defined on a graph: each card in the initial library (before any card gets picked) corresponds to one vertex of that graph, and two vertices of that graph share an edge if the corresponding cards are synergistic. This defines the *library graph*. Now, during each turn of the drafting procedure, first Alice will colour one uncoloured vertex of the library graph red, and then Bob will colour one uncoloured vertex blue. Clearly, a vertex being coloured red (blue resp.) signifies that the corresponding card was picked by Alice (Bob resp.). We now have an important decision to make, namely how to define a metric for the synergy of a deck so that we can decide which player did better during the drafting procedure. We chose to count the number of vertices of the largest connected subgraph of the library graph induced by their colour. This choice defines the rather natural combinatorial game, where both players strive to build the largest connected subgraph of a given graph. We stress here that the game we introduce is not a sufficient model to study the above scenario. We comment more about a possible improvement of the model in the conclusion of this thesis. Nevertheless, this game, which we named the *largest connected subgraph game*, had not been introduced or studied before.

The formal definition of the *largest connected subgraph game* is as follows. The game is played on any graph G between the first player, *Alice*, and the second player, *Bob*. Initially, none of the vertices are coloured. Then, in each round, first Alice colours an uncoloured vertex of G red, and then, Bob colours an uncoloured vertex of G blue. Note that each vertex can only be coloured once and, once coloured, its colour cannot be modified. The game ends when all of the vertices of G have been coloured. The *score* of Alice (Bob resp.) is exactly the order of the largest red (blue resp.) connected subgaph of G by the end of the game. The winner of the game is the player with the highest score at the end of the game. In the case where both players have the same score, the game is a draw. Observe that in this game, Alice and Bob have the same goal: to achieve the largest score. Moreover, they both strive to create connected structures. It follows from these observation that this game is a *connection* and *scoring* game in the same time. The details on these families of games are postponed until Section 1.3.1.

While researching the largest connected subgraph game, it was observed that there are some graphs of order n, for which Alice can always guarantee a score of $\lceil \frac{n}{2} \rceil$, which is the highest score that a player can achieve. In fact, identifying such graphs, said *A-perfect*, turned out to be crucial in order to identify good strategies for both players. Motivated by this observation, we introduced a variation of the largest connected subgraph game, in which Bob's goal becomes to minimise Alice's score, without caring for his own score anymore. This is the *Maker-Breaker largest connected subgraph game*. In particular, the interesting parameter here is defined as the largest integer k such that Alice has a strategy guaranteeing her a score of at least k in the Maker-Breaker largest connected subgraph game played on the given graph G. Section 1.3.2 contains more details on Maker-Breaker games.

1.3.1 Connection and Scoring games

As explained above, in the largest connected subgraph game both players strive to create connected structures. Thus, this game can be placed in the field of study of *connection games*. Several of these games are well-known, such as the game of *Hex* [72]. The game of Hex is played by two players on a rhombus-shaped board tiled by hexagons, with two of the opposing sides of the board coloured red and the other two coloured blue. In each round, the first player colours an uncoloured hexagonal tile red, and then, the second player colours one blue. The first (second, resp.) player wins if they manage to connect the red (blue, resp.) sides of the board with red (blue, resp.) tiles. In the 1950s [73], a variation of the game of Hex, called the *Shannon switching game*, was introduced. In this game, the first player has the goal of connecting two marked vertices in a graph, while the second player wants to make sure this never happens. Traditionally, the players take turns selecting edges of the graph, with the first player winning if there is a path consisting of only the first player's edges between the two marked vertices, but a variant where the players select vertices (and obtain all their incident edges) also exists.

However, not all connection games involve connecting sides of a board or two vertices in a graph. *Havannah*, a board game invented by Christian Freeling that was released in 1981, is one such game, where the players may also win by forming closed loops, with the playing board and the rules being similar to Hex. Connection games tend to be very difficult complexity-wise (one of the reasons they are played and studied), with the majority of them being PSPACE-complete. For example, the generalised Hex, the Shannon switching game on vertices (when players select vertices instead of edges), and the (generalised) Havannah are all PSPACE-complete [109, 67, 39]. That being said, the Shannon switching game on edges is polynomial-time solvable [43]. For more on the complexity of other connection games, see [39]. For more on connection games in general, see [42] for a book on such games.

Another characteristic of the largest connected subgraph game, is that the winner of the game is the player with the highest score, where the score of a player is defined as explained above. Thus, this game can also be placed in the field of study of *scoring games*. The score in these games is an abstract quantity usually measured in an abstract unit called *points*. Players may gain points in a myriad of ways, all depending on the rules of the game. For example, in the orthogonal colouring game on graphs [7], a player's score is equal to the number of coloured vertices in their copy of the graph at the end of the game, *i.e.*, each player gets one point for each coloured vertex in their copy of the graph. Recently, the papers [88, 89, 90] have started to build a general theory around scoring games. There have also been a number of papers on different scoring games of late, such as [47, 63, 96, 114].

1.3.2 Maker-Breaker games

Consider once more the game of Hex. At first glance, there is another similarity between the game of Hex and the largest connected subgraph game, namely that in both games, both players appear to have the same goal, *i.e.*, to be the first player that builds the connected structure required to win the game. However there is a small, but very important difference to be noted here: contrary to the largest connected subgraph game, any game of Hex is a win for Alice if and only if it is a loss for Bob. That is, there is no game of Hex that is a draw. Actually, if we assume that both players play optimally, then Alice is always guaranteed to win. This can be shown by using the famous strategy stealing argument (which we also employ in Chapter 7). In a nutshell, if Bob has a winning strategy then Alice can steal it. That is, after an arbitrary first move, Alice can pretend to be Bob, and follow the Bob's winning strategy. By doing that, Alice can eventually win the game, which contradicts the existence of Bob's winning strategy.

The game of Hex has gathered a lot of interest. We refer the interested reader to [41], a book that gathers various strategies that can be followed during a game of Hex. One of the reasons for which Hex is interesting is that it can also be seen as a game in which Alice tries to connect the red sides of the board with red tiles, while Bob tries to stop her. Such games are known as *Maker-Breaker* games. Actually, both the game of Hex and the Shannon switching game are considered as some of the most famous Maker-Breaker games. In order to understand why Maker-Breaker games have gathered the attention of the scientific community, we have to explain how they can be defined as a subclass of *positional* games.

Positional games are two player games, played on a shared board, which can be abstractedly represented by a set of elements (for example vertices of a graph) X. Now during each turn of a positional game, the players claim previously unclaimed elements of X. In the simplest version, which is of interest for this thesis, each player claims exactly one element of X per turn. To represent the goals of the players, positional games make use of a family $\mathcal{F} = \{A_1, \dots, A_k\}$ of subsets of X, which are usually referred to as the *winning sets* of the *hypergraph* of the game, with each one of the A_is defining a *hyperedge*. Note that the setting of positional games is rather general, as can be attested by the fact that it does not impose any specific winning conditions. One possible winning condition could be "the winner is the first player to capture all the elements of A_i for some $1 \leq i \leq k$". Such games are known as *Maker-Maker* games. Another possible set of winning conditions could be "the first player wins if they capture all the elements of A_i, for some $1 \leq i \leq k$, while the second player wins if, when there are no more unclaimed elements remaining in X, the first player has not won yet". The family of Maker-Breaker games is defined as the positional games that verify this second set of winning conditions, with the first and second players usually being referred to as the *Maker* the *Breaker* respectively. For more details on positional games, the interested reader is invited to see [77].

Returning once more to the game of Hex, it is now formally justifiable that it is a Maker-Breaker game. Indeed, let X be the hexagonal tiles of the board and $\mathcal{F}$ be all the possible "chains" of tiles connecting the red sides of the board. It was also claimed in the first paragraph of this section that any game of Hex is a win for Alice if and only if it is a loss for Bob. This is actually proven in [71]. So the winning condition of Hex can be translated into "Alice wins if she manages to capture one hyperedge of F and Bob wins if, once all the elements of $\mathcal{F}$ are captured, Alice has not won".

The first famous result, which sparked the interest to study Maker-Breaker games, is the so-called Erdős-Selfridge Theorem [65]. Consider any Maker-Breaker game played on X, with $\mathcal{F} = \{A_1, \dots, A_k\}$. According to the Erdős-Selfridge Theorem, if $\Sigma_{i=1}^{k} 2^{-|A_i|} < 1/2$, then the Breaker has a winning strategy. We stress here that the proof of this theorem is not just existential. That is, the proof does not only shows the existence of a winning strategy for the Breaker, but also provides us with one such strategy. Informally, if the conditions of the theorem are satisfied, then it suffices for the Breaker to follow a greedy algorithm, claiming in each turn an element of X that minimises the "worth function" of the remaining unclaimed elements of X. This worth function is well defined, and gets updated during each turn, taking into account the elements that the Maker has claimed.

On the complexity side of things, Schaefer proved in 1978 that determining the outcome of a Maker-Breaker game is PSPACE-complete, even when each of the winning sets in $\mathcal{F}$ has size at most 11 (or exactly 11) [111]. This result was not improved upon until quite recently, when Rahman and Watson proved that determining the outcome of a Maker-Breaker game is PSPACE-complete, even when each of the winning sets in $\mathcal{F}$ has size at most 6 (or exactly 6) [108]. These complexity results are substantial since the problem proven to be PSPACE-complete under the above restrictions, commonly known as POS CNF, is a common problem to reduce from in order to prove PSPACE-hardness, and the size of the largest winning set often has implications on the properties of PSPACE-hard instances of the problem being reduced to.

Apart from general results for Maker-Breaker games, many individual such games have been considered. The following are some of the more notable Maker-Breaker games played on graphs. In particular, in 1978, Chvátal and Erdős introduced the following Maker-Breaker games played on the complete graph K_n: the *Hamiltonicity game*, the *Connectivity game*, and the *Clique game* [55].

In each of these games, X consists of the edges of K_n, while $\mathcal{F}$ consists of each Hamiltonian cycle for the former, each spanning tree for the second, and each clique of a given size for the latter. They notably also introduced *biased* Maker-Breaker games, which are those in which Breaker may instead select multiple elements of X on each of his turns, and the goal is to determine the least number he may select, while still guaranteeing him winning [55]. Note also that the Colouring construction game (presented in Section 1.1) is another example of a Maker-Breaker game. Indeed, this game can be described by a set X which contains exactly the vertices of the given graph, while $\mathcal{F}$ consists of all the proper k-colourings of the graph, for the given k. Clearly, the Maker wins in the colouring construction game if they manage to construct a proper colouring of the given graph, while the Breaker wins if they manage to stop this from happening. Note the small intricacy in the definition of this game, namely that the Breaker is forced to colour vertices while respecting the property that the under-construction colouring should be proper. If this condition was not in place, then the game would be quite trivial (and thus uninteresting), as it would suffice that the Breaker colours a neighbour of an already coloured vertex v with the same colour as that of v, to win the game. Lastly, it is worth mentioning a recently introduced Maker-Breaker game played on graphs called the *Maker-Breaker Domination game*, conceived by Duchêne et al. in 2020 [62], in which X consists of the vertices of the graph, while $\mathcal{F}$ consists of all the dominating sets of the graph.

In the next section we present an overview of the results presented in this thesis, as well as how this thesis is organised.

1.4 Results and layout of this thesis

Chapters 2, 3, 4, 5 and 6 will present our work on proper labellings of graphs. Chapters 7, 8 and 9 will be consecrated to the study of the largest connected subgraph game.

1.4.1 Proper labellings

Chapter 2 presents an overview of some of the tools employed in the first part of this thesis. It is also the chapter in which we present the three conjectures on proper labellings which we were lead to propose through our research. In particular, in Section 2.1, we present these three conjectures, and we comment on the nice interplay that rises between them. Then, in Sections 2.2 and 2.3, we present some tools and important graphs that are going to be useful in the next chapters.

Chapter 3 presents our work on equitable proper labellings. Our results answer some questions that were left open in [18]. In particular, we focus our attention on identifying graphs G such that $\chi_\Sigma(G) = 2 < \overline{\chi_\Sigma}(G) = 3$; let us say that a graph verifying this property is *bad*. We first prove that given a graph G such that $\chi_\Sigma(G) = 2$, it is NP-complete to decide if $\overline{\chi_\Sigma}(G) = 2$ as well. This implies that there are infinitely many bad graphs. We then turn our attention towards bipartite graphs, and prove that once more there exist arbitrarily many bad bipartite graphs. We finally show that, for every $k \geq 3$, every k-regular bipartite graph admits an equitable proper k-labelling. This chapter presents a joint work with J. Bensmail, F. Mc Inerney and N. Nisse, published in [26].

Chapter 4 considers the problem of designing proper labellings which additionally minimise the sum of the labels being used. It would directly follow from the 1-2-3 Conjecture, if it were true, that for every nice graph G, there exists a proper labelling such that the sum of the assigned labels is at most $3|E(G)|$. It turns out that this problem is actually quite different from the 1-2-3 Conjecture. Indeed, we prove that there is no absolute constant k such that for any graph G, the proper labelling of G that minimises the sum of labels being assigned is a k-labelling. We also prove that the problem of designing proper labellings with minimum label sum is NP-hard in general, but solvable in polynomial time for graphs with bounded treewidth and maximum degree, as well as other, simple, families of graphs. We then conjecture that for almost every connected graph G there should be a proper labelling with label sum at most $2|E(G)|$, which we verify for several classes of graphs. In particular, we manage to prove this conjecture for the class of bipartite graphs, and we go on to propose an even stronger conjecture for these graphs, namely that there should always be a proper labelling with label sum at most $\frac{3}{2}|E(G)| + c$ (for some constant c), which we prove for trees. This chapter presents a joint work with J. Bensmail and N. Nisse, published in [29] and presented in [28].

In Chapter 5 we study the problem of designing proper 3-labellings that also minimise the number of edges labelled 3. We prove that, for every $p \geq 0$, there are various graphs needing at least p 3s in their proper 3-labellings. Actually, deciding whether a given graph can be properly 3-labelled with p 3s is NP-complete for every $p \geq 0$. We then focus on classes of 3-chromatic graphs. For various classes of such graphs (cacti, cubic graphs, triangle-free planar graphs, etc.), we prove that there is no $p \geq 1$ such that all their graphs admit proper 3-labellings assigning label 3 to at most p edges. In such cases, we provide lower and upper bounds on the number of 3s needed by any proper 3-labelling of a graph belonging to one of these families. Our investigation leads us to propose a conjecture stating that for any nice graph G, there should exist a proper 3-labelling of G which assigns label 3 on at most $\frac{1}{3}$ of the edges of G. This chapter presents a joint work with J. Bensmail and F. Mc Inerney, published in [22] and presented in [23].

In Chapter 6 we deal with the following question: given a graph G, what is the order of the largest induced subgraph of G that is locally irregular? Or, equivalently, what is the minimum number of vertices that must be deleted from G so that what remains is locally irregular? Let us denote by $\mathrm{I}(G)$ this parameter. We first examine some simple graph families, namely paths, cycles, trees, complete bipartite and complete graphs. We then show that the decision version of the introduced problem is NP-complete, even for restricted families of graphs. We then show that we cannot even approximate an optimal solution within a ratio of $\mathcal{O}(n^{1-\frac{1}{k}})$, where $k \geq 1$ and n is the order the graph, unless P=NP, even when the input graph is bipartite. Then we provide two parameterised algorithms that compute the value $\mathrm{I}(G)$, for a given graph G, each one considering different parameters. The first one considers the size of the solution k and the maximum degree Δ of G while the second one considers the treewidth tw and Δ of G . Therefore, we show that the problem is FPT by both k and tw if the graph has bounded maximum degree Δ. It is natural to wonder if there exists an FPT algorithm parameterised only by k or tw. We answer negatively to this question, by showing that our algorithms are essentially optimal. In particular, we prove that there is no algorithm that computes $\mathrm{I}(G)$ with dependence $f(k)n^{o(k)}$ or $f(tw)n^{o(tw)}$, unless the EXPONENTIAL TIME HYPOTHESIS fails. This chapter presents a joint work with N. Melissinos and T. Triomatis, presented in [69].

1.4.2 Combinatorial games

Chapters 7, 8 and 9 are dedicated to the study of the largest connected subgraph game and its Maker-Breaker version. The largest connected subgraph game is a joint work with J. Bensmail, F. Mc Inerney and N. Nisse, published in [25] and presented in [24]. A report about the Maker-Breaker version, which is a joint work with J. Bensmail, F. Mc Inerney, N. Nisse and N. Oijid, can be found in [27].

In Chapter 7, we give some first results for both games under consideration. Among these results, we prove that Bob can never win the largest connected subgraph game. We also provide some first upper and lower bounds on the largest score that Alice can achieve when playing the Maker-Breaker version of the game on the graph G. We then characterise a class of graphs such that Bob can always guarantee a draw when playing the largest connected subgraph game on a graph belonging to that class. Graphs belonging to this class are called *reflection graphs*. Section 7.3 is consecrated to proving that, given a graph G, determining the outcome of any one of the two games presented in this chapter is PSPACE-complete, even if we assume that G belongs to some restricted families of graphs such as bipartite graphs of diameter 5 for the largest connected subgraph game, and, additionally, split and planar graphs for its Maker-Breaker variation. We also show that identifying reflection graphs is as hard as the GRAPH ISOMORPHISM problem.

Chapter 8 presents our results on some simple families of graphs. First we determine the outcome of the either version of the largest connected subgraph game, when it is played on a graph that is either a path or a cycle. Then, in Section 8.2, we determine the outcome of the largest connected subgraph game when it is played on a cograph. It is worth mentioning here that determining the outcome of the Maker-Breaker version when playing on cographs is far less arduous than determining the outcome of the largest connected subgraph game when playing on these graphs. Thus, in Section 8.3, we also manage to provide a linear-time algorithm that decides the outcome of the Maker-Breaker version when playing on $(q, q-4)$-graphs, a family which naturally generalises cographs.

Chapter 9 deals with A-perfect graphs. Recall that a graph G is said to be A-perfect if Alice can always create a single connected component when playing the Maker-Breaker largest connected subgraph game on G. We begin by focusing our attention on the problem of identifying such graphs that are also regular. In particular, in Section 9.2.1, we prove that for every $d \geq 4$, there exist arbitrarily large d-regular graphs that are A-perfect. On the contrary, in Section 9.2.2, we prove that any 3-regular A-perfect graph is of order at most 16. Finally, we provide sufficient conditions for a graph to be A-perfect, either in regards to its degrees, or to its size.

Variations of proper labellings

CHAPTER 2

Three new conjectures, useful tools and graphs

In Section 2.1 of this chapter we formally define the three problems that are going to be the main focus of Chapters 3, 4 and 5. These are optimisation problems, which are about finding proper labellings that also verify some additional constraints, and define three natural parameters which we study. We also propose three new conjectures on the upper bound of these new parameters, and explain how the study of each one of them leads us naturally to the study of the others. Finally, in Sections 2.2 and 2.3 we provide some general tools that are going to be useful later on, as well as the description of some important graphs.

2.1 Definition of three variations and their interplay

In this section we formally introduce the problems on proper labellings that are treated in Chapters 3, 4 and 5.

Let $G = (V, E)$ be a graph. Recall that a k-labelling ℓ of a graph G, is a function that assigns a number from $\{1, \dots, k\}$ on each edge of G; these numbers will be referred to as labels. Any k-labelling defines a vertex-colouring c_ℓ of G, where each vertex $v \in V$ receives as colour $c_\ell(v) = \Sigma_{u \in N(v)} \ell(uv)$. This $c_\ell(v)$ is said to be the colour of v induced by ℓ. If the colouring c_ℓ is a proper vertex-colouring of G, ℓ is said a proper k-labelling of G. The parameter $\chi_\Sigma(G)$ is used to denote the minimum k such that G admits a proper k-labelling. The parameter χ_Σ was initially introduced in [81], in which the authors propose the following conjecture:

1-2-3 Conjecture. *If G is a nice* * *graph, then* $\chi_\Sigma(G) \leq 3$.

$*$. Recall that a graph is said nice if it does not contain K_2 as a connected component

Chapter 3 regards the problem of finding equitable proper labellings, that is proper labellings in which each label is assigned more or less the same number of times. Formally:

Definition 2.1.1. *Let $G = (V, E)$ be a nice graph and ℓ a k-labelling of G. For every label α assigned by ℓ (meaning that there exists at least one edge of G labelled α by ℓ), let $\mathrm{nb}_\ell(\alpha)$ be the number of edges of G being labelled α by ℓ. That is,*

$$\mathrm{nb}_\ell(\alpha) = |\{e \in E : \ell(e) = \alpha\}|.$$

A k-labelling ℓ is said to be equitable if for every two distinct labels α, β assigned by ℓ, the values $\mathrm{nb}_\ell(\alpha)$ and $\mathrm{nb}_\ell(\beta)$ differ by at most 1. That is,

$$\max_{i,j} |\mathrm{nb}_\ell(i) - \mathrm{nb}_\ell(j)| \leq 1,$$

for $1 \leq i < j \leq k$. The parameter $\overline{\chi_\Sigma}(G)$ is used to denote the minimum k such that G admits an equitable proper k-labelling. This notion was initially introduced and studied in [18].

It follows from the results in [34] that for any nice graph G, the parameter $\overline{\chi_\Sigma}(G)$ is well defined. In particular, we know that $\overline{\chi_\Sigma}(G)$ is upperly bounded by a function of the number of edges of G. Nevertheless, to date, there is no constant upper bound for $\overline{\chi_\Sigma}(G)$. The authors of [18] have investigated several aspects of equitable proper labellings, most of which are about the relationship between $\chi_\Sigma(G)$ and $\overline{\chi_\Sigma}(G)$ for a given graph G. Note that K_4 is the only nice graph G (that we know of) verifying $\overline{\chi_\Sigma}(G) > 3$. Their research leads us to wonder about the plausibility of an equitable version of the 1-2-3 Conjecture:

Conjecture 2.1.2 (Equitable 1-2-3 Conjecture)**.** *If G is a nice graph other than K_4, then $\overline{\chi_\Sigma}(G) \leq 3$.*

Consider now any nice graph G, other than K_4, and let ℓ be an equitable proper 3-labelling of G. Observe that the number of edges labelled 1 by ℓ is about the same as the number of edges labelled 3 by ℓ. It follows that the sum of labels assigned to the edges of G by ℓ, is about $2|E(G)|$. Note that this sum is, in general, much smaller than what is suggested by the immediate upper bound on that value, which follows from the result that $\chi_\Sigma(G) \leq 5$ [79], *i.e.*, that the sum of the labels assigned by ℓ is at most $5|E(G)|$. This leads us directly to the next variation of proper labellings, considered in Chapter 4, namely the problem of finding proper labellings that minimise the sum of the assigned labels. Formally:

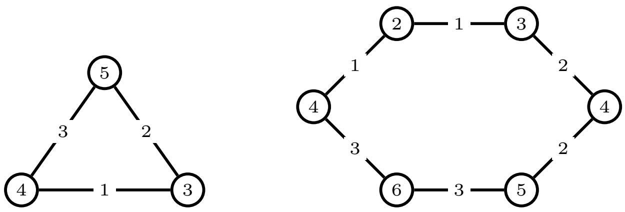
Definition 2.1.3. *For a labelling ℓ of a nice graph $G = (V, E)$, let $\sigma(\ell)$ denote the sum of labels assigned to the edges of G by ℓ. That is,*

$$\sigma(\ell) = \sum_{e \in E} \ell(e).$$

For any $k \geq \chi_\Sigma(G)$, denote by $\mathrm{mE}_k(G)$ the minimum value of $\sigma(\ell)$ over all proper k-labellings ℓ of G. That is,

$$\mathrm{mE}_k(G) = \min \{\sigma(\ell) : \ell \text{ is a proper } k\text{-labelling of } G\}.$$

Finally, let $\mathrm{mE}(G) = \min\{\mathrm{mE}_k(G) : k \geq \chi_\Sigma(G)\}$.



(a) K_3 (b) C_6

Figure 2.1 – Proper 3-labellings of K_3 and C_6.

Observe that the 1-2-3 Conjecture, if true, would imply that, for every nice graph G, $\mathrm{mE}(G) \leq \mathrm{mE}_3(G) \leq 3|E(G)|$. A natural question to wonder is whether there exist graphs G for which $\mathrm{mE}(G)$ is close to this theoretical upper bound of $3|E(G)|$. The fact that we are not aware of graphs G with $\chi_\Sigma(G) = 3$ needing a lot of 3s in all proper 3-labellings, as well as Conjecture 2.1.2, leads us to suspect that even the following conjecture might be true:

Conjecture 2.1.4. *For every nice graph $G = (V, E)$, $\mathrm{mE}(G) \leq 2|E|$.*

Clearly, Conjecture 2.1.4 holds true for all nice graphs G with $\chi_\Sigma(G) \leq 2$. Experimentation via computer programs led us to observe that, actually, it might even be true that, when considering connected graphs, the equality $\mathrm{mE}(G) = 2|E(G)|$ holds if and only if G is K_3 or C_6. Note that $\chi_\Sigma(K_3) = \chi_\Sigma(C_6) = 3$. This is a direct consequence of the following observation, which is useful when dealing with graphs that contain adjacent vertices of degree 2, and is going to remain important throughout the first part of this thesis.

Observation 2.1.5. *Let G be a graph containing two adjacent vertices v_2 and v_3 of degree 2. Let v_1 be the other neighbour of v_2, and let v_4 be the other neighbour of v_3 (possibly $v_1 = v_4$). Then, by any proper labelling ℓ of G, $\ell(v_1v_2) \neq \ell(v_3v_4)$.*

Proof. Since, by any proper labelling ℓ of G, it holds that $c_\ell(v_2) = \ell(v_1v_2) + \ell(v_2v_3)$, $c_\ell(v_3) = \ell(v_2v_3) + \ell(v_3v_4)$, and $c_\ell(v_2) \neq c_\ell(v_3)$, then $\ell(v_1v_2) \neq \ell(v_3v_4)$. □

Actually, it also follows from Observation 2.1.5 that any proper 3-labelling ℓ of C_6 will be such that $\mathrm{nb}_\ell(1) = \mathrm{nb}_\ell(2) = \mathrm{nb}_\ell(3) = 2$. In Figure 2.1 we illustrate the only possible 3-labelling of K_3 and one proper 3-labelling of C_6. However, these cases are very particular, due to the small number of edges these two graphs have.

Note also that we do manage to prove Conjecture 2.1.4 for the case of nice bipartite graphs (see Section 4.4.2). This leads us to propose an even stronger conjecture for the case of bipartite graphs, which we manage to prove for the case of trees:

Conjecture 2.1.6. *There is an absolute constant $c \geq 1$ such that, for every nice connected bipartite graph G, $\mathrm{mE}_2(G) \leq \frac{3}{2}|E| + c$.*

Informally, a labelling verifying Conjecture 2.1.6 assigns labels 1 and 2 to almost half the edges of G each, and label 3 to the rest; the constant c that appears in the conjecture corresponds exactly to the number of edges labelled 3.

Note also that the number of edges labelled 3 by the labellings proposed in the previous paragraph, is rather small. This falls in line with a long-standing belief of this field, *i.e.*, that proper

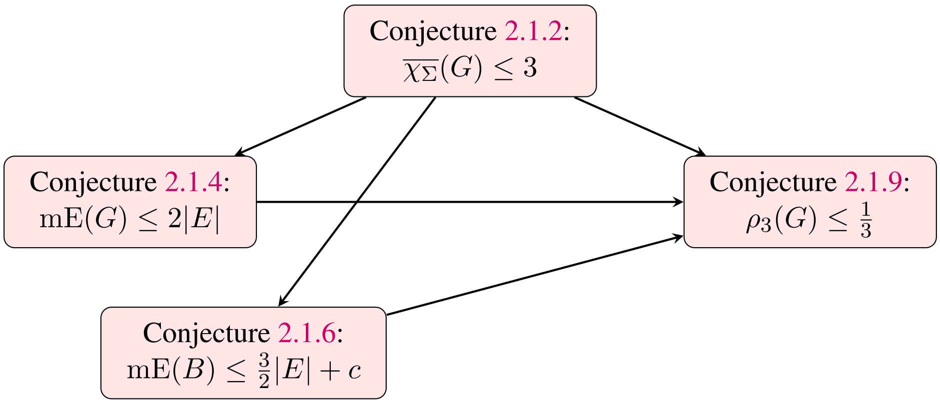


Figure 2.2 – A schematic representation of the interplay between the conjectures proposed in this thesis. The graph G is assumed to be any nice graph different from K_4, and B is assumed to be any nice connected bipartite graph. The set E represents the edges of the corresponding graph. The arrows are not to be interpreted as "implies", but rather "leads to the study of".

3-labellings do not actually require "a lot" of edges labelled 3. It is exactly to better understand and quantify this "a lot" that we introduce the final variation of proper labellings considered in this thesis in Chapter 5, namely the problem of constructing proper 3-labellings that also minimise the number of edges labelled 3. Formally:

Definition 2.1.7. *For any nice graph G, we denote by $\mathrm{mT}(G)$ the minimum number of edges assigned label 3 by a proper 3-labelling of G. That is,*

$$\mathrm{mT}(G) = \min\{\mathrm{nb}_\ell(3) : \ell \text{ is a proper 3-labelling of } G\}.$$

We extend this parameter mT to classes $\mathcal{F}$ of graphs by defining $\mathrm{mT}(\mathcal{F})$ as the maximum value of $\mathrm{mT}(G)$ over the members G of $\mathcal{F}$. Clearly, $\mathrm{mT}(\mathcal{F}) = 0$ for every class $\mathcal{F}$ of graphs admitting proper 2-labellings (i.e., $\chi_\Sigma(G) \leq 2$ for every $G \in \mathcal{F}$).

Given a graph class $\mathcal{F}$, we are interested in determining whether $\mathrm{mT}(\mathcal{F}) \leq p$ for some $p \geq 0$. From this perspective, for every $p \geq 0$, we denote by $\mathcal{G}_p$ the class of graphs G with $\mathrm{mT}(G) = p$. For convenience, we also define $\mathcal{G}_{\leq p} = \mathcal{G}_0 \cup \cdots \cup \mathcal{G}_p$.

Our research suggests that, for several graph classes $\mathcal{F}$, there is no $p \geq 0$ such that $\mathcal{F} \subset \mathcal{G}_{\leq p}$. For such a class, we want to know whether the proper 3-labellings of their members require assigning label 3 many times, with respect to their number of edges. We study this aspect through the following terminology.

Definition 2.1.8. *For a nice graph G, we define*

$$\rho_3(G) = \mathrm{mT}(G)/|E(G)|.$$

We extend this ratio to a class of graphs $\mathcal{F}$ by setting

$$\rho_3(\mathcal{F}) = \max\{\rho_3(G) : G \in \mathcal{F}\}.$$

We are thus interested in determining bounds on $\rho_3(\mathcal{F})$ for some graph classes $\mathcal{F}$ whose members admit proper 3-labellings, and, more generally speaking, in how large this ratio can be.

Note that this is similar to considering how large $\rho_3(G)$ can be for a given graph G. Also, notice that graphs G of small size with $\chi_\Sigma(G) = 3$ are more likely to require many (compared to $|E(G)|$) 3s to be properly 3-labelled, resulting in these graphs having "large" $\rho_3(G)$ (meaning a value close to 1). Through an exhaustive search, it is easy to see that, among the sample of small connected graphs (*e.g.*, of order at most 6), the maximum ratio ρ_3 is exactly $1/3$, which is attained by K_3 and C_6. At the moment, these are the worst graphs we know of, which leads us to raising the following conjecture.

Conjecture 2.1.9. *If G is a nice connected graph, then $\rho_3(G) \leq 1/3$.*

It is worth adding that Conjecture 2.1.9 can be seen as a weaker version of Conjecture 2.1.2, in the sense that if the latter were to be proven correct then the former would also follow. To sum up, Conjecture 2.1.2 leads directly to both Conjectures 2.1.4 and 2.1.9, while Conjecture 2.1.4 also leads to Conjecture 2.1.9. So there is a very nice correlation between the conjectures proposed in our work, which we schematically depict in Figure 2.2.

2.2 Useful techniques

In this section we present a set of techniques that are going to be useful in the following chapters.

2.2.1 Proper labellings through stable sets

We now present a version of the proof that any nice 3-chromatic graph admits proper 3-labellings. This theorem, initially presented in [81], is fundamental in this field. Furthermore, we make use of the approach followed in the proof of this theorem in Chapters 4 and 5.

The following proof is based on carefully labelling a particular subgraph of the given 3-chromatic nice graph G. Since G is 3-chromatic, it follows that G is not bipartite. Thus there exists an odd-length cycle C in G. Let H be a subgraph of G constructed as follows. Start from $H = C$. Then, until $V(H) = V(G)$, repeatedly choose a vertex $v \in V(G) \setminus V(H)$ such that there exists a vertex $u \in V(H)$ with $uv \in E(G)$, and add the edge uv to H. In the end, H is a connected spanning subgraph of G containing only one cycle, C, which is of odd length. Then, we have $|E(H)| = |V(G)|$. This graph H is denoted as an *odd unicyclic (connected) spanning subgraph* of G.

Theorem 2.2.1 ([81]). *If G is a nice connected 3-chromatic graph, then $\chi_\Sigma(G) \leq 3$.*

Proof. Let $\phi : V(G) \to \{0, 1, 2\}$ be a proper 3-vertex-colouring of G. In what follows, our goal is to construct a 3-labelling ℓ of G such that $c_\ell(v) \equiv \phi(v) \bmod 3$ for every vertex $v \in V(G)$, thus making ℓ proper. Note that, aiming at vertex colours modulo 3, we can instead assume that ℓ assigns labels $0, 1, 2$. To obtain such a labelling, we start from ℓ assigning label 2 to all edges of G. We then modify ℓ iteratively until all vertex colours are as desired modulo 3. Recall that a *walk* in a graph is a path in which vertices and/or edges might be repeated. A walk is said *closed* if it starts and ends at a same vertex. We say a path is *even* (*odd*, resp.) if it consists of an even (odd, resp.) number of edges.

Let H be an odd unicyclic spanning subgraph of G. As long as G has a vertex v with $c_\ell(v) \not\equiv \phi(v) \bmod 3$, we apply the following procedure. Choose $W = (v, v_1, \ldots, v_n, v)$, a closed walk of odd length in G starting and ending at v, and going through edges of H only. This walk is sure to exist. Indeed, consider, in H, a (possibly empty) path P from v to the closest vertex u of C (if v lies on C, then $u = v$ and P has no edge). Then, the closed walk $vPuCuPv$ is a possible W. We then follow the consecutive edges of W, starting from v and ending at v, and, going along, we apply $+2, -2, +2, -2, \dots, +2$ (modulo 3) to the labels assigned by ℓ to the traversed edges. As a result, note that $c_\ell(x)$ is not altered modulo 3 for every vertex $x \neq v$, while $c_\ell(v)$ is incremented by 1 modulo 3. If $c_\ell(v) \equiv \phi(v) \bmod 3$, then we are done with v. Otherwise, we repeat this switching procedure once again, so that v fulfils that property.

Eventually, we get $c_\ell(v) \equiv \phi(v) \bmod 3$ for every $v \in V(G)$, meaning that ℓ is proper. All that remains to be done is to replace the label of the edges labelled 0 by ℓ by the label 3. The resulting labelling is a proper 3-labelling of G and thus $\chi_\Sigma(G) \leq 3$. □

The importance of the above proof is twofold. On the one hand, it can easily be modified in order to achieve proper 3-labellings with additional constraints, as is done in Chapter 5. On the other hand, one can replicate this proof even for graphs of chromatic number $k \geq 3$, and design proper (not necessarily 3-)labellings verifying the desired properties, as is done in Chapter 4.

On an even more general level, the core idea behind the proof of Theorem 2.2.1 is to take advantage of a partition of the vertices of the given graph G into p stable sets. Given such a partition $S_0, \dots, S_{p-1}$, we can try to construct a labelling ℓ such that for each vertex $v \in V(G)$, if $v \in S_i$ then $c_\ell(v)$ is congruent to i (modulo p). In the case of 3-chromatic graphs, this construction is achieved through an odd unicyclic spanning subgraph of G. But this is not the only way such a labelling could be constructed. The important thing to note here is that constructing ℓ in such a way is extremely helpful, as it guarantees that the labelling is proper, without having to check if the induced colouring is indeed proper. This idea is behind many arguments that are fairly common in this field (see *e.g.* [16, 32, 46, 81, 94, 95, 116]). Let us present one more example of this idea. Note that the following lemma is going to prove useful in Chapter 4.

Lemma 2.2.2 ([46]). *Let $G = (U, V, E)$ be a connected bipartite graph. If at least one of $|U|$ or $|V|$ is even, then $\chi_\Sigma(G) \leq 2$.*

Proof. Assume w.l.o.g. that $|U|$ is even. We will construct a 2-labelling ℓ such that

1. for every $u \in U$, $c_\ell(u)$ is odd and
2. for every $v \in V$, $c_\ell(v)$ is even.

We start by having ℓ assign label 2 on all the edges of G. At this stage, only the vertices of V verify the above condition. To make sure that also the vertices of U verify the above condition, we pair the vertices of U two by two. Let u_1, u_2 be one such pair. Since G is connected and bipartite, there exists an even-length path P from u_1 to u_2. Following P, we swap the labels of the edges we encounter (so label 1 becomes 2 and *vice versa*). This procedure only changes the parity of $c_\ell(u_1)$ and $c_\ell(u_2)$. Repeating this procedure for all the pairs of vertices of U results in the desired 2-labelling. □

As a final example of using stable sets to produce proper labellings, we mention here the case of nice trees.

Proposition 2.2.3. *Let $T = (V, E)$ be a nice tree. Then $\chi_\Sigma(T) \leq 2$. Furthermore, a proper 2-labelling of T can be constructed in linear time.*

Proof. We provide a procedure that constructs a proper 2-labelling of T in linear time. The following observation is of crucial importance, and will appear many times throughout this thesis:

Observation 2.2.4. *Let G be a nice graph containing at least one vertex v such that $d(v) = 1$, and let u be the sole neighbour of v. Then, for any k-labelling ℓ of G, $c_\ell(v) < c_\ell(u)$.*

Let us now root T on a vertex r, and let $L_0, \dots, L_p$ be the levels of T. That is, a vertex v belongs in L_i, for $0 \leq i \leq p$, if its distance from r is exactly i. We will construct a 2-labelling ℓ such that for every non-leaf vertex $v \in V$, if $v \in L_i$, then $c_\ell(v)$ has the same parity as i. Clearly, a labelling that respects this condition is proper as any two adjacent non-leaf vertices of a rooted tree belong in different levels, and thus their colours induced by ℓ have different parities. Note that the parity of the induced colours of the leaves of T is unspecified. This does not create any colouring conflict thanks to Observation 2.2.4.

The construction of ℓ is as follows. We start by having ℓ assign label 1 on all the edges of T. Now, we consider the vertices of T in a breadth first search fashion, starting from r. In this search, if we encounter a non-leaf vertex v whose induced colour does not have the correct parity, we change the label of uv from 1 to 2, where u is any vertex amongst the children of v. Note that this modification does not affect the colour of the parent of v. At the end of the search, ℓ will verify the desired condition. $\square$

Since determining $\chi_\Sigma(G)$ can be done efficiently when G is a tree (thanks to Proposition 2.2.3 and because determining if a graph is locally irregular can be done in linear time), it makes sense to wonder whether calculating $\chi_\Sigma(G)$ can be done efficiently when G is a graph of bounded tree-width.

2.2.2 Dynamic programming on nice tree-decompositions

Two of the algorithms we present in Chapters 4 and 6 are dynamic programming algorithms, which produce proper labellings of graphs that are "close to being trees". These graphs are formally defined through the notion of a *tree-decomposition*:

Definition 2.2.5. *Given a graph $G = (V, E)$, a tree-decomposition of G is a pair $(T, \mathcal{X})$ such that $T = (V(T), E(T))$ is a tree and $\mathcal{X} = \{X_t \subseteq V \mid t \in V(T)\}$ is a family of subsets (called bags) of vertices of G such that:*

- *$V = \bigcup_{t \in V(T)} X_t$;*
- *for every $uv \in E$, there exists $t \in V(T)$ with $u, v \in X_t$; and*
- *for every $v \in V$, the subset $\{t \in V(T) \mid v \in X_t\}$ induces a subtree of T.*

The width of $(T, \mathcal{X})$ is equal to $\max_{t \in V(T)} |X_t| - 1$ and the treewidth $\mathrm{tw}(G)$ is the minimum width of a tree-decomposition of G.

In order to render the above definition, as well as our dynamic programming algorithms presented in Chapters 4 and 6, more comprehensible, let us give an intuition on how the above notion can be used algorithmically by presenting the following toy example. We consider the problem of deciding if for a given graph G, $\chi(G) \leq 3$. Recall that this problem is NP-complete [74]. We will now present a dynamic programming algorithm that solves this problem in polynomial time on graphs of bounded treewidth.

Theorem 2.2.6 ([10])**.** *Let G be a graph and $(T, \mathcal{X})$ be a rooted (with root r) tree-decomposition of G of width $k - 1$, for some fixed $k \geq 2$. Deciding if $\chi(G) \leq 3$ can be done in polynomial time.*

Sketch of the proof. We will consider one by one the nodes of $V(T)$ in a bottoms-up fashion, starting from the leaves of T and progressing towards r. Before explaining how we treat each node, we need a final definition. Let t_1 and t_2 be any two nodes of T. Moreover, let c_1 (c_2 resp.) be a colouring of the vertices of X_{t_1} (X_{t_2} resp.). We will say that c_1 *agrees* with c_2 if for every vertex $v \in X_{t_1} \cap X_{t_2}$, $c_1(v) = c_2(v)$. If $X_{t_1} \cap X_{t_2} = \emptyset$ then every pair of colourings for t_1 and t_2 agree with each other.

Let t be a leaf node of T. Since the width of the tree-decomposition is $k-1$, we know that t contains at most k vertices. For each vertex v of X_t, consider the three colours that c could assign to v. There are at most 3^k possible colourings for the vertices of X_t. Among all these colourings, keep the ones that are proper colourings of $G[X_t]$. If no such colouring exists, then $\chi(G) > 3$ and the algorithm stops.

Assume now that t is a non-leaf node of T, and that we have already treated all the nodes $t_1, \ldots, t_d$ that are the children of t. For each $1 \leq i \leq d$, let $C^i = \{c_1^i, c_2^i, \ldots, c_p^i\}$ be the list of all the colourings already stored for t_i (which are proper colourings of $G[X_{t_i}]$ for every i). Note that $p \leq 3^k$. Consider once more the list C of all the at most 3^k possible colourings of the vertices of X_t that are also proper colourings of $G[X_t]$. Among all the colourings of C, we only keep the ones that agree with at least one colouring in C^i for every $1 \leq i \leq d$, and delete the rest. If after concluding this procedure C is empty, then $\chi(G) > 3$ and the algorithm stops.

If we manage to successfully treat all the nodes of T, then $\chi(G) \leq 3$. Moreover, we have to treat at most $\mathcal{O}(n)$ such nodes (where n is the order of G), and each node requires a time which is exponential only in k. Therefore the algorithm runs in polynomial (in respect to n) time. $\diamond$

Note that in the algorithm presented above, it would have been practical if the given tree-decomposition was such that any two neighbouring (in T) bags differed by at most one vertex. This idea is captured by the notion of *nice* tree-decompositions. Formally:

Definition 2.2.7. *A tree-decomposition $(T, \mathcal{X})$ is nice [38] if T is rooted in $r \in V(T)$ and every node $t \in V(T)$ is exactly of one of the following four types:*

1. ***Leaf:*** *t is a leaf of T and $|X_t| = 1$.*
2. ***Introduce:*** *t has a unique child t' and there exists $v \in V$ such that $X_t = X_{t'} \cup \{v\}$.*
3. ***Forget:*** *t has a unique child t' and there exists $v \in V$ such that $X_{t'} = X_t \cup \{v\}$.*
4. ***Join:*** *t has exactly two children t', t'' and $X_t = X_{t'} = X_{t''}$.*

It is well known that every graph $G = (V, E)$ admits a nice tree-decomposition $(T, \mathcal{X})$ rooted in $r \in V(T)$, that has width equal to $\mathrm{tw}(G)$, $|V(T)| = \mathcal{O}(|V|)$ and $X_r = \{\emptyset\}$ [38].

We finish this section with some additional notation which will prove useful for the algorithms on graphs of bounded treewidth presented in Chapters 4 and 6. Let $(T, \mathcal{X})$ be a rooted tree-decomposition (with root r) of G and $t \in V(T)$. A subtree of T induced by t and its descendants is denoted as T_t and the corresponding subgraph of G, *i.e.*, the graph induced by $\cup_{t' \in V(T_t)} X_{t'}$, is denoted by G_t (clearly, $G_r = G$). Let us also set $V_t = V(G_t)$ for convenience. Notice that $X_t \subseteq V_t$. Finally, for every $v \in V_t$, we denote by $N_t(v)$ the neighbourhood of v in G_t, that is $N_t(v) = \{u \in V_t : uv \in E(G_t)\}$.

2.3 Useful graphs

In this section we present a family of graphs and a set of gadgets that are going to be useful throughout the next chapters.

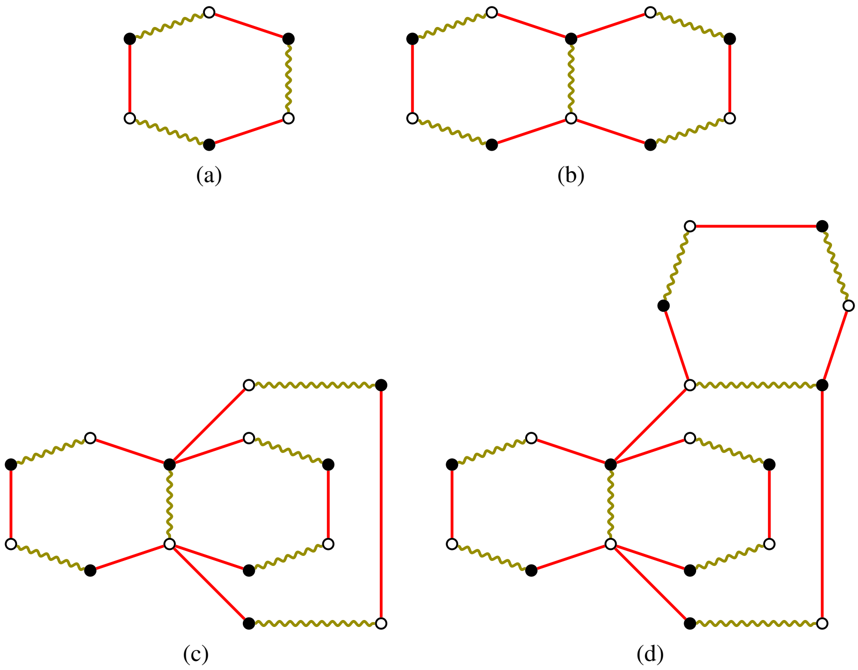


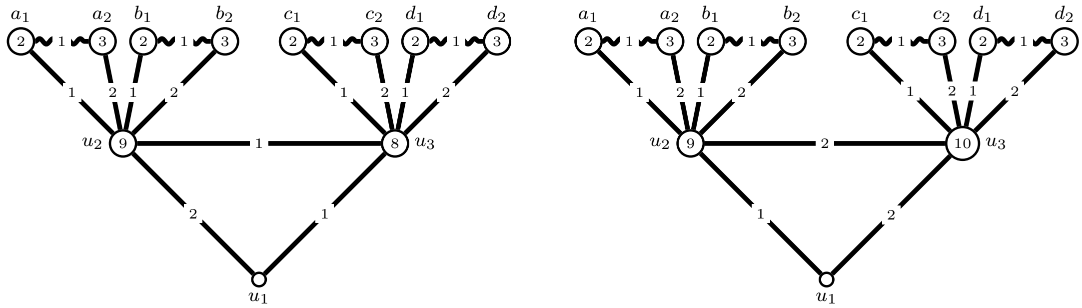
Figure 2.3 – Constructing an odd multi-cactus through several steps, from the red-olive C_6 (a). Red-olive paths with length at least 5 congruent to 1 modulo 4 are being repeatedly attached onto olive edges through steps (b) to (d). Solid edges are red edges. Wiggly edges are olive edges.

2.3.1 Odd multi-cacti

One of the most well understood families of graphs in the context of proper labellings, is that of bipartite graphs. It was shown already in [81], via a modification of the proof of Lemma 2.2.2, that nice bipartite graphs G verify the 1-2-3 Conjecture. Note also that some bipartite graphs G verify $\chi_\Sigma(G) = 3$ (for example C_6).

Thus, connected nice bipartite graphs can be classified into three classes $\mathcal{B}_1$, $\mathcal{B}_2$ and $\mathcal{B}_3$ where, for each $i \in \{1, 2, 3\}$, the set $\mathcal{B}_i$ contains exactly the connected bipartite graphs G with $\chi_\Sigma(G) = i$. Clearly, $\mathcal{B}_1$ contains exactly the bipartite graphs that are locally irregular. Recall that, in general, it is NP-complete to decide if a given graph G verifies $\chi_\Sigma(G) \leq 2$. Note that this complexity result does not hold for bipartite graphs. So, the natural question becomes whether we can efficiently distinguish the bipartite graphs belonging in $\mathcal{B}_2$ from those belonging in $\mathcal{B}_3$. We have already presented some small steps towards this direction, namely that a nice bipartite graph G that is not locally irregular, belongs in $\mathcal{B}_2$ if G is a tree or at least one of its bipartition classes is of even size. The first real step towards answering this question appears in [94], where it was shown that if a bipartite graph G is 3-connected, then $G \in \mathcal{B}_2$. The definitive answer to the question of classifying bipartite graphs came when the graphs of $\mathcal{B}_3$ were characterized by Thomassen, Wu and Zhang in [116], who proved that these graphs are exactly the so-called *odd multi-cacti*. As we are going to provide results for these graphs in Chapters 3 and 4, we give their definition here.

Definition 2.3.1. *A graph G is an odd multi-cactus if it can be obtained at any step of the following procedure (see Figure 2.3 for an illustration):*



(a) $\{c_{\ell_1}(u_2), c_{\ell_1}(u_3)\} = \{8, 9\}$. (b) $\{c_{\ell_2}(u_2), c_{\ell_2}(u_3)\} = \{9, 10\}$.

Figure 2.4 – The only proper 2-labellings ℓ_1 (left) and ℓ_2 (right) of the T_2 gadget, used in the construction illustrated in Figure 2.5. The induced colours for each labelling are represented as integers in the corresponding vertices. Vertex u_1 is called the *root* of the gadget. Wiggly edges are edges that could be labelled either 1 or 2.

- *Start from a cycle with length at least* 6*, congruent to* 2 *modulo* 4*, whose edges are properly coloured with red and olive (i.e., no two adjacent edges have the same colour).*
- *Consider an olive edge uv and join u and v via a new path of length at least* 5*, congruent to* 1 *modulo* 4*, whose edges are properly coloured with red and olive, where both the edge incident to u and the edge incident to v are red.*

It is worth mentioning that odd multi-cacti are 2-degenerate and 2-connected. Also, they are bipartite, and both of their parts have odd size. It can also be noted that for every olive edge uv, we have $d(u) = d(v)$, and no two olive edges share ends.

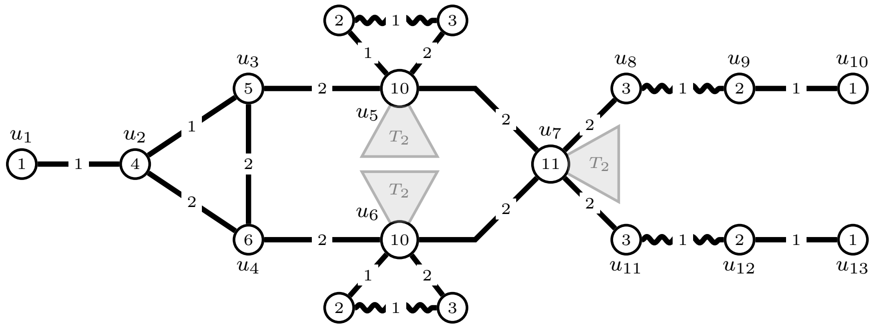
Theorem 2.3.2 ([116])**.** *Let G be a nice connected bipartite graph. Then $\chi_\Sigma(G) = 3$ if and only if G is an odd multi-cactus.*

2.3.2 Useful gadgets

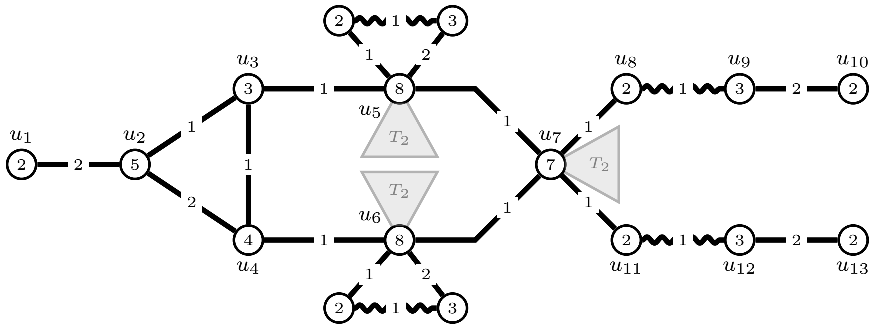
In this section we present a graph admitting proper 2-labellings, that also has a very useful property: it contains three *particular edges* such that, by any proper 2-labelling of that graph, these particular edges are assigned the same label. What is even more useful, is that many copies of this graph can be combined, by correctly identifying their respective particular edges, with the resulting graph having a multitude of new particular edges, all of which must receive the same label by every proper 2-labelling.

Definition 2.3.3. *The graph illustrated in Figure 2.5 is called the spreading gadget and denoted by $G^{\curlywedge}$. In $G^{\curlywedge}$, the parts denoted by a T_2 correspond to the graph illustrated in Figure 2.4; let us call T_2 the triangle gadget. The edge u_1u_2 will be referred to as the* input *and the edges u_9u_{10} and $u_{12}u_{13}$ as the* outputs *of $G^{\curlywedge}$.*

This gadget was initially introduced in [20] where it was shown that any proper 2-labelling ℓ of $G^{\curlywedge}$ verifies $\ell(u_1u_2) = \ell(u_9u_{10}) = \ell(u_{12}u_{13})$. Furthermore, the different proper 2-labellings of $G^{\curlywedge}$ have a very specific behaviour in regards to how many edges are labelled by 1 and by 2, which is going to prove useful in our study of equitable proper labellings, in Chapter 3, and proper



(a) Input is labelled 1.



(b) Input is labelled 2.

Figure 2.5 – The only proper 2-labellings of the spreading gadget $G^{\curlywedge}$. A triangle marked as "T_2" indicates that a copy of the T_2 gadget (depicted in Figure 2.4) is attached via its root vertex. That is, u_5 (resp., u_6 and u_7) is identified to the root of one copy of T_2. The induced colours for each labelling are represented as integers in the corresponding vertices. Wiggly edges are edges that could be labelled either 1 or 2.

labellings that also minimise the sum of labels being used, in Chapter 4. Since our results will take advantage of this particular behaviour, we include here a more detailed version of the proof of the following three theorems, initially presented in [20]:

Theorem 2.3.4. *In any proper* 2*-labelling* ℓ *of* T_2*:*

- $\{\ell(u_1u_2), \ell(u_1u_3)\} = \{1,2\}$*; furthermore:*
  - *Either* $\{c_\ell(u_2), c_\ell(u_3)\} = \{8,9\}$*, and* $\mathrm{nb}_\ell(1)$ *can be any value in* $\{6,\dots,10\}$*;*
  - *or* $\{c_\ell(u_2), c_\ell(u_3)\} = \{9,10\}$*, and* $\mathrm{nb}_\ell(1)$ *can be any value in* $\{5,\dots,9\}$.

Proof. Let ℓ be a proper 2-labelling of T_2. Since $c_\ell(a_1) \neq c_\ell(a_2)$, we have, say, $\ell(a_1u_2) = 1$ and $\ell(a_2u_2) = 2$. Note that whatever $\ell(a_1a_2)$ is, no conflict involving a_1 (or a_2) and u_2 can arise, due to the larger degree of u_2. These arguments also apply around the b_i's, c_i's, and d_i's. In particular, the labels of the four edges joining u_2 to the a_i's and b_i's bring 6 to the colour of u_2, and similarly the labels of the four edges joining u_3 to the c_i's and d_i's bring 6 to the colour of u_3. Now, since $c_\ell(u_2) \neq c_\ell(u_3)$, we have, say, $\ell(u_1u_2) = 1$ and $\ell(u_1u_3) = 2$. Then, no conflict involving u_2 and u_3 can arise, no matter whether u_2u_3 is labelled 1 or 2. In the first case, we get $(c_\ell(u_2), c_\ell(u_3)) = (8,9)$, while we get $(c_\ell(u_2), c_\ell(u_3)) = (9,10)$ in the second case.

The parts of the statement dealing with $\mathrm{nb}_\ell(1)$ hold, essentially, because each of the edges a_1a_2, b_1b_2, c_1c_2, d_1d_2, and u_2u_3 can freely be assigned any label in $\{1,2\}$ by ℓ. $\square$

Theorem 2.3.5. *$G^{\curlywedge}$ verifies the following:*
- — $|E(G^{\curlywedge})| = 89$.
- — *In any proper 2-labelling ℓ of $G^{\curlywedge}$, we have $\ell(u_1u_2) = \ell(u_9u_{10}) = \ell(u_{12}u_{13})$.*
- — *There exist both proper 2-labellings ℓ of $G^{\curlywedge}$ where $\ell(u_1u_2) = 1$, and proper 2-labellings ℓ of $G^{\curlywedge}$ where $\ell(u_1u_2) = 2$.*
- — *In any proper 2-labelling ℓ of $G^{\curlywedge}$ where $\ell(u_1u_2) = 1$:*
  - — *$c_\ell(u_2) = 4$;*
  - — *$c_\ell(u_9)$ and $c_\ell(u_{12})$ can be any value in $\{2, 3\}$; furthermore:*
    - — *if $c_\ell(u_9) = c_\ell(u_{12}) = 2$, then $\mathrm{nb}_\ell(1)$ can be any value in $\{35, \dots, 56\}$;*
    - — *if $c_\ell(u_9) = c_\ell(u_{12}) = 3$, then $\mathrm{nb}_\ell(1)$ can be any value in $\{33, \dots, 54\}$;*
    - — *if $\{c_\ell(u_9), c_\ell(u_{12})\} = \{2, 3\}$, then $\mathrm{nb}_\ell(1)$ can be any value in $\{34, \dots, 55\}$.*
- — *In any proper 2-labelling ℓ of $G^{\curlywedge}$ where $\ell(u_1u_2) = 2$:*
  - — *$c_\ell(u_2) = 5$;*
  - — *$c_\ell(u_9)$ and $c_\ell(u_{12})$ can be any value in $\{3, 4\}$; furthermore:*
    - — *if $c_\ell(u_9) = c_\ell(u_{12}) = 3$, then $\mathrm{nb}_\ell(1)$ can be any value in $\{35, \dots, 56\}$;*
    - — *if $c_\ell(u_9) = c_\ell(u_{12}) = 4$, then $\mathrm{nb}_\ell(1)$ can be any value in $\{33, \dots, 54\}$;*
    - — *if $\{c_\ell(u_9), c_\ell(u_{12})\} = \{3, 4\}$, then $\mathrm{nb}_\ell(1)$ can be any value in $\{34, \dots, 55\}$.*

Proof. Consider ℓ a proper 2-labelling of $G^{\curlywedge}$. We first note that we have $\ell(u_3u_5) = \ell(u_4u_6)$. Indeed, suppose to the contrary that, *e.g.*, $\ell(u_3u_5) = 1$ and $\ell(u_4u_6) = 2$ holds. Since there are two copies of T_2 attached to u_5, by Theorem 2.3.4, the colour of u_5 is $7 + \ell(u_5u_7)$ and it is adjacent to a vertex with colour 9 (in T_2). Similarly, because of the two copies of T_2 attached to u_6, the colour of u_6 is $8 + \ell(u_6u_7)$ and it is adjacent to a vertex with colour 9 (in T_2). Then, we must have $\ell(u_5u_7) = 1$ and $\ell(u_6u_7) = 2$, so that $c_\ell(u_5) = 8$ and $c_\ell(u_6) = 10$. We also know that a neighbour of u_7 from the graph T_2 attached to it has colour 9, and that this graph T_2 provides 3 to the colour of u_7 by Theorem 2.3.4. Then, u_7 has colour $6 + \ell(u_7u_8) + \ell(u_7u_{11})$, and the two edges u_7u_8 and u_7u_{11} must be labelled (with 1 or 2) in such a way that the colour of u_7 does not meet any value in $\{8, 9, 10\}$, which is impossible.

Now suppose $\ell(u_1u_2) = 1$, and consider the edges u_2u_3 and u_2u_4 (see Figure 2.5(a) for an illustration). First, if $\ell(u_2u_3) = \ell(u_2u_4)$, then note that ℓ cannot be proper according to the arguments above since we would need to have $\ell(u_3u_5) \neq \ell(u_4u_6)$ since $c_\ell(u_3) \neq c_\ell(u_4)$. Thus, $\ell(u_2u_3) = 1$ and $\ell(u_2u_4) = 2$ without loss of generality, and $c_\ell(u_2) = 4$. Note that, if $\ell(u_3u_4) = 1$, then we necessarily get that $c_\ell(u_3)$ or $c_\ell(u_4)$ is equal to $c_\ell(u_2)$ since we need $\ell(u_3u_5) = \ell(u_4u_6)$. Thus, $\ell(u_3u_4) = 2$. We then have $\ell(u_3u_5) = 2$ since $c_\ell(u_3) \neq c_\ell(u_2)$, and also $\ell(u_4u_6) = 2$ since $c_\ell(u_4) \neq c_\ell(u_3)$ (and because $\ell(u_4u_6) = \ell(u_3u_5)$ by the arguments above).

According to the arguments above, we have $\ell(u_5u_7) = \ell(u_6u_7) = 2$. By the same arguments and since $c_\ell(u_5) = c_\ell(u_6) = 10$, we have $\ell(u_7u_8) = \ell(u_7u_{11}) = 2$. Then, $\ell(u_9u_{10}) = \ell(u_{12}u_{13}) = 1$ to avoid conflicts. Thus, assuming the input of $G^{\curlywedge}$ is labelled 1, then its two outputs are also labelled 1. A similar case analysis yields an analogous conclusion when $\ell(u_1u_2) = 2$, see Figure 2.5(b).

Let us conclude by pointing out that, in the proper labellings of $G^{\curlywedge}$ mentioned above, the only edges for which the assigned label can freely be either 1 or 2 are the edges u_8u_9, $u_{11}u_{12}$, four edges in each of the four copies of T_2 attached to u_5 and u_6, and five edges in the copy of T_2 attached to u_7. As pointed out earlier, all other edges must (up to symmetry) receive a particular

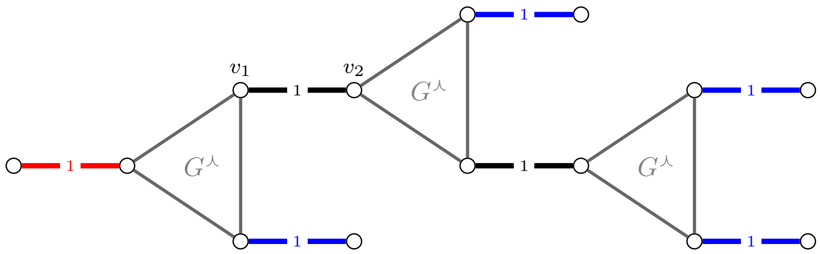


Figure 2.6 – An example of the generator gadget G_4. The triangles are meant to represent the copies of $G^{\curlywedge}$. The red edge is the input of G_4, as well as the input of G_2, the first copy of $G^{\curlywedge}$ that was used in the construction. The blue edges are the outputs of G_4. The edge v_1v_2 corresponds to the edge u_9u_{10} of G_2 (following the naming convention of Figure 2.5), which coincides with the edge u_1u_2 of the copy of $G^{\curlywedge}$ that was used to construct G_3. The label 1 is used to demonstrate the property shown in Theorem 2.3.7.

label in $\{1, 2\}$ as soon as that of u_1u_2 is fixed. It is then easy to check that the parts of the statement dealing with $\mathrm{nb}_\ell(1)$ are true. □

The last gadget we need is the following.

Definition 2.3.6. *The generator gadget G_m is a graph with $m \geq 2$ outputs and one input, which is obtained from $m-1$ spreading gadgets as follows. For $m = 2$, $G_2 = G^{\curlywedge}$. For $m = 3$, the generator gadget G_3 with three outputs is obtained by plugging two copies H_1 and H_2 of the spreading gadget $G^{\curlywedge}$ along any output of H_1 and the input of H_2. The input of G_3 is then the input of H_1 and the three outputs of G_3 are the second (unplugged) output of H_1 and the two outputs of H_2. For $m > 3$, the generator gadget G_m with m outputs is obtained by plugging a copy G of the generator gadget G_{m-1} with $m-1$ outputs and a new copy H of the spreading gadget $G^{\curlywedge}$ along one output of G and the input of H. The input of G_m is then the input of G and the m outputs of G_m are the remaining $m-2$ (unplugged) outputs of G and the two outputs of H.*

Figure 2.6 illustrates the generator gadget G_4.

Theorem 2.3.7. *G_m verifies the following, for every $m \geq 3$:*
- $|E(G_m)| = 88m - 87$.
- *In any proper 2-labelling of G_m, the input and m outputs are assigned the same label.*
- *There exist both proper 2-labellings of G_m where the input is assigned label 1, and proper 2-labellings of G_m where the input is assigned label 2.*
- *In any proper 2-labelling ℓ of G_m assigning label 1 to the input, $\mathrm{nb}_\ell(1) \in \{32m - 31, \dots, 55m - 54\}$.*
- *In any proper 2-labelling ℓ of G_m assigning label 2 to the input, $\mathrm{nb}_\ell(1) \in \{33m - 33, \dots, 56m - 56\}$.*

Proof. This follows essentially from Theorem 2.3.5, since G_m is made up of $m-1$ copies of $G^{\curlywedge}$. In particular, any proper 2-labelling ℓ of G_m induces one of each of its $m-1$ underlying $G^{\curlywedge}$'s. As pointed out in the statement of Theorem 2.3.5, vertices identified through the plugging operation cannot get in conflict.

The part of the statement dealing with $\mathrm{nb}_\ell(1)$ is essentially because, in each copy of $G^{\curlywedge}$ in G_m, there are 23 edges that can freely be set to 1 or 2 (four edges in four attached copies of

T_2, five edges in the last attached copy of T_2, and two edges adjacent to the outputs). Assuming the input of G_m is labelled 1 by ℓ, according to Theorem 2.3.5 in each of the copies of $G^{\curlywedge}$ the number of edges that can be assigned label 1 essentially ranges from 33 to 56. Thus, in G_m, the number of edges that can be assigned label 1 ranges from $33(m-1)-(m-2)=32m-31$ to $56(m-1)-(m-2)=55m-54$. The computation is similar when the input of G_m is assigned label 2, the only difference is that copies of $G^{\curlywedge}$ do not share edges labelled 1. $\square$

Chapter 3

Equitable proper labellings

In this chapter, we consider equitable proper labellings of graphs, which were introduced by Baudon, Pilśniak, Przybyło, Senhaji, Sopena, and Woźniak.
We provide results regarding some open questions about equitable proper labellings. Via a hardness result, we first prove that there exist infinitely many graphs for which more labels are required in the equitable version than in the non-equitable version. This remains true in the bipartite case. We finally show that, for every $k \geq 3$, every k-regular bipartite graph admits an equitable proper k-labelling.
This chapter presents a joint work with J. Bensmail, F. Mc Inerney and N. Nisse, published in [26].

This chapter focuses on equitable proper labellings, which are proper labellings in which all labels must be assigned about the same number of times. Recall that a labelling ℓ is called equitable if, for every two distinct labels α, β assigned by ℓ, meaning that there exists at least one edge of the graph labelled α and one labelled β, the number of edges assigned label α differs from the number of edges assigned label β by at most 1. The smallest k such that G admits an equitable proper k-labelling is denoted by $\overline{\chi_\Sigma}(G)$. This parameter $\overline{\chi_\Sigma}$ is defined for every nice graph. The formal definition of these notions can be found in Definition 2.1.1.

Let us start this chapter by briefly summarising the main results on equitable proper labellings, found in [18]:

— For nice forests F, we always have $\overline{\chi_\Sigma}(F) = \chi_\Sigma(F) \leq 2$.

— For nice complete bipartite graphs $K_{n,m}$, we always have $\overline{\chi_\Sigma}(K_{n,m}) = \chi_\Sigma(K_{n,m}) \leq 2$, except for the peculiar case of $K_{3,3}$ which verifies $2 = \chi_\Sigma(K_{3,3}) < \overline{\chi_\Sigma}(K_{3,3}) = 3$.
— For nice complete graphs K_n, we always have $\overline{\chi_\Sigma}(K_n) = \chi_\Sigma(K_n) = 3$, except for the peculiar case of K_4 which verifies $3 = \chi_\Sigma(K_4) < \overline{\chi_\Sigma}(K_4) = 4$.

At this point, the previous results lead to a number of interesting questions. Is K_4 the only graph G with $\overline{\chi_\Sigma}(G) > 3$? Are graphs G with $\chi_\Sigma(G) < \overline{\chi_\Sigma}(G)$ rare? Can the difference between $\chi_\Sigma(G)$ and $\overline{\chi_\Sigma}(G)$ be arbitrarily large? In general, could it be that if G is a nice graph other than K_4, then $\overline{\chi_\Sigma}(G) \leq 3$ (appearing in this thesis as Conjecture 2.1.2)?

A few more results partially answering some of these questions can be found in Senhaji's thesis [113]. In particular:
— Senhaji proved that $\overline{\chi_\Sigma}(G) = \chi_\Sigma(G) \leq 3$ holds for a certain number of graphs G, including nice paths, nice cycles, some Theta graphs, and some Cartesian products of graphs.
— Using computer programs, he came up with four cubic bipartite graphs G verifying $2 = \chi_\Sigma(G) < \overline{\chi_\Sigma}(G) = 3$.
— For particular cubic bipartite graphs G, such as Hamiltonian ones, he proved that $\overline{\chi_\Sigma}(G) \leq 2$.

In this chapter, we provide results towards some of the questions above. In particular, we investigate the existence of graphs G with $\chi_\Sigma(G) < \overline{\chi_\Sigma}(G)$. In Section 3.1, we first prove that there exist infinitely many such graphs. This is obtained through proving that the problem of deciding whether $\overline{\chi_\Sigma}(G) = 2$ holds for a given graph G with $\chi_\Sigma(G) = 2$ is NP-complete. We then investigate, in Section 3.2, the same question for bipartite graphs. We exhibit operations establishing that there exist infinitely many bipartite graphs G with $\chi_\Sigma(G) < \overline{\chi_\Sigma}(G)$. We also prove that for every bipartite graph G with $\chi_\Sigma(G) = 3$, *i.e.*, odd multi-cacti, we have $\overline{\chi_\Sigma}(G) = 3$. In Section 3.3, we finally provide a result on equitable proper labellings of regular bipartite graphs, showing that $\overline{\chi_\Sigma}(G) \leq k$ holds for every such k-regular graph ($k \geq 3$). In particular, we have $\overline{\chi_\Sigma}(G) \leq 3$ for every cubic bipartite graph G.

3.1 Hardness result

This section is devoted to the proof that the problem of deciding whether $\overline{\chi_\Sigma}(G) = 2$ holds for a given graph G with $\chi_\Sigma(G) = 2$ is NP-complete.

The reduction in the proof of our main result below will be mostly obtained by *plugging* several *gadgets*, some of which are already introduced in Section 2.3.2, with specific properties together. More precisely, some of our gadgets will have specific pendent edges (*i.e.*, with exactly one of their ends being of degree 1) being their *inputs* or *outputs*. Given two disjoint gadgets G and H where e is an output of G and f is an input of H, by *plugging* G and H (along e and f) we mean identifying e and f together. More precisely, if $e = xy$ and $f = uv$ with y and v being the vertices of degree 1 of e and f respectively, then identifying e and f means identifying x and v, and y and u respectively.

High-level description. Before describing the explicit construction in our reduction, let us first give some intuition about its desired structure, as well as the gadgets that will be used.

The most important gadget for our construction is the generator gadget G_m, presented in Section 2.3.2, Definition 2.3.6. The most convenient feature of the gadget G_m comes from Theorem 2.3.7, which lets us generate arbitrarily many pendent edges with the same label by a proper

2-labelling. This, for instance, permits to make the colours of some vertices grow by a similar amount, or, as will be illustrated later, to forbid some values as vertex colours.

The generator gadget G_m has several downsides, however. A first one is that we, *a priori*, do not know whether its input and outputs will be labelled 1 or 2 by a proper 2-labelling. A second one is that G_m does not comply well with equitability, in the sense that, generally speaking, it admits both proper 2-labellings highly favouring the number of assigned 1s, and proper 2-labellings highly favouring the number of assigned 2s (recall Theorem 2.3.7).

To overcome these points, we will use two additional types of gadgets we will call the *initiator gadget* I_k and the *corrector gadget* C (to be formally defined in the subsections below), in the following way. The initiator gadget I_k will be used to introduce a large imbalance in favour of one of the two labels by any proper 2-labelling. By that, we mean an imbalance that is so big that even all the labelling freedom we have in G_m will not allow to close the gap between the 1s and the 2s. To make sure that the whole graph does admit equitable proper 2-labellings, however, we will add several copies of the corrector gadget C. The most important property of this gadget is that, in terms of equitability, its behaviour regarding label 1 and label 2 is far from symmetric. By that, we mean, that the possibilities C grants highly depend on the label assigned to its input by a proper 2-labelling. If this label is 1, then we can both favour the number of assigned 1s or favour the number of assigned 2s. On the contrary, if this label is 2, then for sure the number of assigned 2s is favoured.

By properly plugging an initiator gadget I_k (for a well chosen value of k) and corrector gadgets C onto the generator gadget G_m, we can, in particular, make sure that the outputs of G_m are all labelled 1 by an equitable proper 2-labelling of the whole graph. This is because, by a proper 2-labelling assigning label 2 to the outputs, the initiator gadget I_k would introduce a huge imbalance in favour of the number of assigned 2s, that is so huge that it cannot be caught up by the labelling freedom of G_m and the copies of the corrector gadget C.

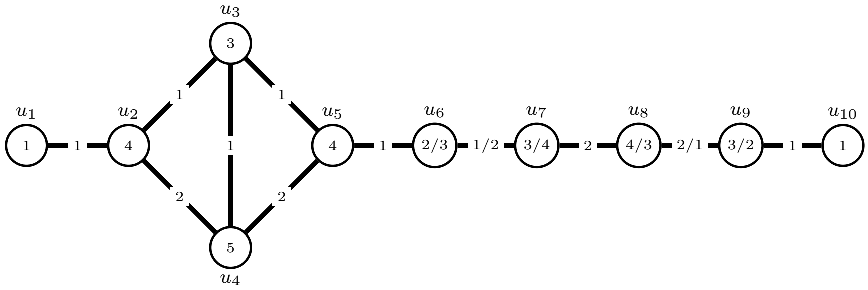
Once we know that the input and all outputs of G_m must be assigned label 1 by any equitable proper 2-labelling, the forcing mechanisms in the whole graph then become much easier to track, and it then becomes easier to design an equivalence with finding a solution to the problem from which we are reducing.

3.1.1 Initiator gadget

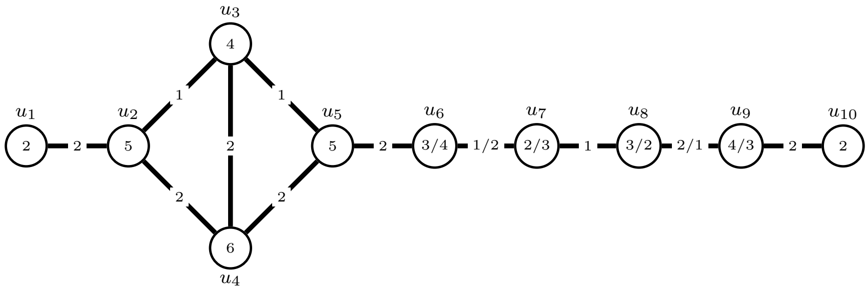
The first gadget we need is the *diamond* D depicted in Figure 3.1. Here and further, for every gadget introduced through a figure, we deal with its vertices and edges following the notation from that figure. The input of D is the edge u_1u_2, while the output of D is the edge u_9u_{10}. The properties of interest of D are the following:

Theorem 3.1.1. *D verifies the following:*

- $|E(D)| = 11$.
- *In any proper* 2*-labelling ℓ of D, we have $\ell(u_1u_2) = \ell(u_9u_{10})$.*
- *There exist both proper* 2*-labellings ℓ of D where $\ell(u_1u_2) = 1$, and proper* 2*-labellings ℓ of D where $\ell(u_1u_2) = 2$.*
- *In any proper* 2*-labelling ℓ of D where $\ell(u_1u_2) = 1$:*
  - $c_\ell(u_2) = 4$;
  - $c_\ell(u_9)$ *can be any value in* $\{2, 3\}$;
  - $\mathrm{nb}_\ell(1) = 7$ *and* $\mathrm{nb}_\ell(2) = 4$.
- *In any proper* 2*-labelling ℓ of D where $\ell(u_1u_2) = 2$:*



(a) $\ell(u_1u_2) = 1$.



(b) $\ell(u_1u_2) = 2$.

Figure 3.1 – The diamond gadget D. The values in each vertex v are the possible colours of $c_\ell(v)$ by a proper 2-labelling ℓ of D.

— $c_\ell(u_2) = 5$*;*
— $c_\ell(u_9)$ *can be any value in* $\{3, 4\}$*;*
— $\mathrm{nb}_\ell(1) = 4$ *and* $\mathrm{nb}_\ell(2) = 7$.

Proof. Let ℓ be a proper 2-labelling of D. Assume $\ell(u_1u_2) = 1$.

If $\ell(u_2u_3) = \ell(u_2u_4) = 1$, then $c_\ell(u_2) = 3$. Since $c_\ell(u_3) \neq c_\ell(u_4)$, we have $\ell(u_3u_5) = 1$ and $\ell(u_4u_5) = 2$, or *vice versa*. Since $c_\ell(u_3) \neq c_\ell(u_2)$, we have $\ell(u_3u_4) = 2$. This gives $c_\ell(u_3) = 4$ and $c_\ell(u_4) = 5$. Now we note that no matter what $\ell(u_5u_6)$ is, we necessarily get $c_\ell(u_5) \in \{4, 5\} = \{c_\ell(u_3), c_\ell(u_4)\}$, a contradiction. So we cannot have $\ell(u_2u_3) = \ell(u_2u_4) = 1$.

If $\ell(u_2u_3) = \ell(u_2u_4) = 2$, then $c_\ell(u_2) = 5$. Again, since $c_\ell(u_3) \neq c_\ell(u_4)$, we have, say, $\ell(u_3u_5) = 1$ and $\ell(u_4u_5) = 2$. Since $c_\ell(u_4) \neq 5$, we have $\ell(u_3u_4) = 2$, which gives $c_\ell(u_3) = c_\ell(u_2) = 5$, a contradiction. Thus, we cannot have $\ell(u_2u_3) = \ell(u_2u_4) = 2$.

If, say, $\ell(u_2u_3) = 1$ and $\ell(u_2u_4) = 2$, then $c_\ell(u_2) = 4$. Assume first that $\ell(u_3u_4) = 2$. In that case, since $c_\ell(u_3) \neq c_\ell(u_2)$, we have $\ell(u_3u_5) = 2$, and thus $c_\ell(u_3) = 5$. Since $c_\ell(u_4) \neq c_\ell(u_3)$, we have $\ell(u_4u_5) = 2$, and thus $c_\ell(u_4) = 6$. Now, we note that no matter what $\ell(u_5u_6)$ is, we have $c_\ell(u_5) \in \{5, 6\} = \{c_\ell(u_3), c_\ell(u_4)\}$, a contradiction. So, we have $\ell(u_3u_4) = 1$. Since $c_\ell(u_3)$ and $c_\ell(u_4)$ are different from $c_\ell(u_2)$, we have $\ell(u_3u_5) = 1$ and $\ell(u_4u_5) = 2$, which gives $c_\ell(u_3) = 3$ and $c_\ell(u_4) = 5$. Now, since $c_\ell(u_5) \notin \{3, 5\} = \{c_\ell(u_3), c_\ell(u_4)\}$, we have $\ell(u_5u_6) = 1$, which gives $c_\ell(u_5) = 4$. By Observation 2.1.5, we then have $\ell(u_7u_8) \neq \ell(u_5u_6)$ and $\ell(u_9u_{10}) \neq \ell(u_7u_8)$, and thus $\ell(u_7u_8) = 2$ and $\ell(u_9u_{10}) = 1$. By the same argument, we have $\ell(u_6u_7) \neq \ell(u_8u_9)$, and both ways are possible. Indeed, if on the one hand $\ell(u_6u_7) = 1$ and $\ell(u_8u_9) = 2$, then $c_\ell(u_6) = 2$, $c_\ell(u_7) = 3$, $c_\ell(u_8) = 4$, and $c_\ell(u_9) = 3$. If on the other hand $\ell(u_6u_7) = 2$ and $\ell(u_8u_9) = 1$, then $c_\ell(u_6) = 3$, $c_\ell(u_7) = 4$, $c_\ell(u_8) = 3$, and $c_\ell(u_9) = 2$.

According to all these arguments, we have that $\mathrm{nb}_\ell(1) = 7$ while $\mathrm{nb}_\ell(2) = 4$. Also, depending on whether $\ell(u_6u_7) = 1$ and $\ell(u_8u_9) = 2$, or $\ell(u_6u_7) = 2$ and $\ell(u_8u_9) = 1$, the value of $c_\ell(u_9)$ can be any one in $\{2, 3\}$. See Figure 3.1(a) for an illustration of the resulting ℓ.

These arguments can be mimicked the exact same way when $\ell(u_1u_2) = 2$. In particular, we have $\ell(u_1u_2) = \ell(u_9u_{10}) = 2$, $\mathrm{nb}_\ell(1) = 4$ while $\mathrm{nb}_\ell(2) = 7$, and $c_\ell(u_9)$ can be any value in $\{3, 4\}$. See Figure 3.1(b) for an illustration. □

The *initiator gadget* I_k of length $k \geq 2$ has one input and one output, and is obtained from k diamond gadgets as follows. For $k = 2$, the initiator gadget I_2 of length 2 is obtained by plugging two copies D_1 and D_2 of the diamond gadget D along the output of D_1 and the input of D_2. The input of I_2 is then the input of D_1 and the output of I_2 is then the output of D_2. For $k > 2$, the initiator gadget I_k of length k is obtained by plugging a copy G of the initiator gadget I_{k-1} of length $k - 1$ and a new copy H of the diamond gadget D along the output of G and the input of H. The input of I_k is then the input of G and the output of I_k is then the output of H.

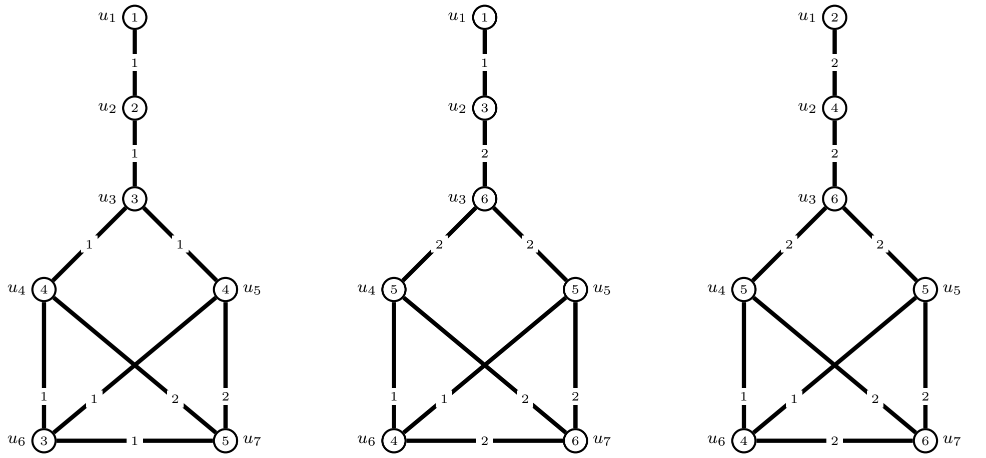
Theorem 3.1.2. *I_k verifies the following, for every $k \geq 2$:*
- $|E(I_k)| = 10k + 1$.
- *In any proper 2-labelling of I_k, the input and output are assigned the same label.*
- *There exist both proper 2-labellings of I_k where the input is assigned label 1, and proper 2-labellings of I_k where the input is assigned label 2.*
- *In any proper 2-labelling ℓ of I_k where the input is assigned label 1:*
  - *$c_\ell(v)$ can be any value in $\{2, 3\}$, where v denotes the degree-2 vertex of the output of I_k;*
  - $\mathrm{nb}_\ell(1) = 6k + 1$ *and* $\mathrm{nb}_\ell(2) = 4k$.
- *In any proper 2-labelling ℓ of I_k where the input is assigned label 2:*
  - *$c_\ell(v)$ can be any value in $\{3, 4\}$, where v denotes the degree-2 vertex of the output of I_k;*
  - $\mathrm{nb}_\ell(1) = 4k$ *and* $\mathrm{nb}_\ell(2) = 6k + 1$.

Proof. This follows mainly from the fact that I_k is made up of k copies of the diamond gadget D plugged one after another, and that the diamond gadget D has all of the properties described in Theorem 3.1.1. In particular, a proper 2-labelling of I_k induces a proper 2-labelling of the k copies of the diamond gadget D in it. Specifically, it can be checked that no conflict can arise around the inputs and outputs that were identified. Also, for a proper 2-labelling of I_k assigning label $\alpha \in \{1, 2\}$ to the input, in each copy of the diamond gadget D label α must be assigned to seven edges while the other label must be assigned to four edges. Due to how the copies of D were plugged, we deduce that $7k - (k - 1) = 6k + 1$ edges of I_k must be assigned label α, while $4k$ edges must be assigned the other label. □

3.1.2 Corrector gadget

The *corrector gadget* C is the graph depicted in Figure 3.2. The input of C is the edge u_1u_2, while C has no output. Its interesting properties are the following:

Theorem 3.1.3. *C verifies the following:*
- $|E(C)| = 9$.
- *There exist both proper 2-labellings ℓ of C where $\ell(u_1u_2) = 1$, and proper 2-labellings ℓ of C where $\ell(u_1u_2) = 2$.*



(a) $\ell(u_1u_2) = 1$, $\mathrm{nb}_\ell(1) = 7$, $\mathrm{nb}_\ell(2) = 2$.
(b) $\ell(u_1u_2) = 1$, $\mathrm{nb}_\ell(1) = 3$, $\mathrm{nb}_\ell(2) = 6$.
(c) $\ell(u_1u_2) = 1$, $\mathrm{nb}_\ell(1) = 2$, $\mathrm{nb}_\ell(2) = 7$.

Figure 3.2 – The corrector gadget C. The values in each vertex v are the possible colours of $c_\ell(v)$ by a proper 2-labelling ℓ of C.

- *In any proper* 2*-labelling* ℓ *of* C *where* $\ell(u_1u_2) = 1$*:*
  - $c_\ell(u_2) \in \{2, 3\}$*;*
  - *either* $\mathrm{nb}_\ell(1) = 7$ *and* $\mathrm{nb}_\ell(2) = 2$*, or* $\mathrm{nb}_\ell(1) = 3$ *and* $\mathrm{nb}_\ell(2) = 6$.
- *In any proper* 2*-labelling* ℓ *of* C *where* $\ell(u_1u_2) = 2$*:*
  - $c_\ell(u_2) = 4$*;*
  - $\mathrm{nb}_\ell(1) = 2$ *and* $\mathrm{nb}_\ell(2) = 7$.

Proof. Let ℓ be a proper 2-labelling of C. Because u_6 and u_7 both have degree 3, we have that $c_\ell(u_6), c_\ell(u_7) \in \{3, 4, 5, 6\}$. Furthermore, $c_\ell(u_6) \neq c_\ell(u_7)$, and we cannot have $\{c_\ell(u_6), c_\ell(u_7)\} = \{3, 6\}$. We consider all of the remaining possibilities for $\{c_\ell(u_6), c_\ell(u_7)\}$ in what follows.

Assume $c_\ell(u_6) = 3$ and $c_\ell(u_7) = 4$. Then, all three edges incident to u_6 must be labelled 1, while we have, say, $\ell(u_7u_4) = 1$ while $\ell(u_7u_5) = 2$. Then, we note that, whatever $\ell(u_4u_3)$ is, we have $c_\ell(u_4) \in \{3, 4\} = \{c_\ell(u_6), c_\ell(u_7)\}$, a contradiction. The case where $\ell(u_7u_4) = 2$ while $\ell(u_7u_5) = 1$ is symmetric with the colour of u_5 coming into conflict with that of u_6 or u_7 instead. Thus, we cannot have $\{c_\ell(u_6), c_\ell(u_7)\} = \{3, 4\}$.

Assume $c_\ell(u_6) = 3$ and $c_\ell(u_7) = 5$. Again, all three edges incident to u_6 must be labelled 1, while we have $\ell(u_7u_4) = \ell(u_7u_5) = 2$. Now, since $c_\ell(u_4)$ and $c_\ell(u_5)$ are different from $5 = c_\ell(u_7)$, we have $\ell(u_4u_3) = \ell(u_5u_3) = 1$. This gives $c_\ell(u_4) = c_\ell(u_5) = 4$. Now, since $c_\ell(u_3)$ is different from $4 = c_\ell(u_4) = c_\ell(u_5)$, we have $\ell(u_3u_2) = 1$. Then $c_\ell(u_3) = 3$, and, since $c_\ell(u_2) \neq c_\ell(u_3)$, we have $\ell(u_2u_1) = 1$, which yields $c_\ell(u_2) = 2$. This is the labelling depicted in Figure 3.2(a).

Assume $c_\ell(u_6) = 4$ and $c_\ell(u_7) = 5$. First, assume $\ell(u_6u_7) = 1$. Then we have, say, $\ell(u_6u_4) = 1$ and $\ell(u_6u_5) = \ell(u_7u_4) = \ell(u_7u_5) = 2$ (the case where $\ell(u_6u_4) = 2$ and $\ell(u_6u_5) = 1$ is symmetric). Note now that whatever $\ell(u_4u_3)$ is, we have $c_\ell(u_4) \in \{4, 5\} = \{c_\ell(u_6), c_\ell(u_7)\}$, a contradiction. Then, assume $\ell(u_6u_7) = 2$. Then we have $\ell(u_6u_4) = \ell(u_6u_5) = 1$ and, say, $\ell(u_7u_5) = 1$ and $\ell(u_7u_4) = 2$ (the case where $\ell(u_7u_5) = 2$ and $\ell(u_7u_4) = 1$ is symmetric). Again, note that whatever $\ell(u_4u_3)$ is, we have $c_\ell(u_4) \in \{4,5\} = \{c_\ell(u_6), c_\ell(u_7)\}$, a contradiction. Thus, we cannot have $\{c_\ell(u_6), c_\ell(u_7)\} = \{4,5\}$

Assume $c_\ell(u_6) = 4$ and $c_\ell(u_7) = 6$. Then, all three edges incident to u_7 must be labelled 2, while we have $\ell(u_6u_4) = \ell(u_6u_5) = 1$. Now, since $c_\ell(u_4)$ and $c_\ell(u_5)$ are different from $4 = c_\ell(u_6)$, we have $\ell(u_4u_3) = \ell(u_5u_3) = 2$. This gives $c_\ell(u_4) = c_\ell(u_5) = 5$. Now, since $c_\ell(u_3)$ is different from $5 = c_\ell(u_4) = c_\ell(u_5)$, we have $\ell(u_3u_2) = 2$. Then, $c_\ell(u_3) = 6$, and note that u_1u_2 can correctly be assigned either of the labels 1 and 2. In the first case, we get the labelling depicted in Figure 3.2(b), in which $c_\ell(u_2) = 3$. In the second case, we get the labelling depicted in Figure 3.2(c), in which $c_\ell(u_2) = 4$.

Assume $c_\ell(u_6) = 5$ and $c_\ell(u_7) = 6$. Then, all three edges incident to u_7 must be labelled 2, while we have, say, $\ell(u_6u_4) = 1$ while $\ell(u_6u_5) = 2$ (the case where $\ell(u_6u_4) = 2$ while $\ell(u_6u_5) = 1$ is symmetric). Then, we note that, whatever $\ell(u_5u_3)$ is, we have $c_\ell(u_5) \in \{5,6\} = \{c_\ell(u_6), c_\ell(u_7)\}$, a contradiction. Thus, we cannot have $\{c_\ell(u_6), c_\ell(u_7)\} = \{5,6\}$. □

3.1.3 Main result

We are now ready for the main result of this section.

Theorem 3.1.4. *Given a graph G with $\chi_\Sigma(G) = 2$, deciding if $\overline{\chi_\Sigma}(G) = \chi_\Sigma(G)$ is* NP*-complete.*

Proof. The problem is clearly in NP, so we focus on proving it is NP-hard. We do it by reduction from the MONOTONE CUBIC 1-IN-3 SAT problem, which is NP-hard according to [100]. An instance of this problem is a 3CNF formula F in which every clause $C_j = (x_1 \vee x_2 \vee x_3)$ contains exactly three distinct variables (not negated) and every variable x_i belongs to exactly three distinct clauses. The question is whether there is a 1-in-3 truth assignment to the variables of F, *i.e.*, a truth assignment such that every clause has exactly one true variable. Given F, we construct, in polynomial time, a graph G such that F admits a 1-in-3 truth assignment ϕ if and only if G admits an equitable proper 2-labelling ℓ.

The construction of G is as follows. Let us start from the cubic bipartite graph G_F modelling the structure of the 3CNF formula F. That is, for every variable x_i of F we add a *variable vertex* v_i to the graph, for every clause C_j of F we add a *clause vertex* c_j to the graph, and, whenever a variable x_i belongs to a clause C_j in F, we add the *formula edge* v_ic_j to the graph.

We also add a generator gadget G_μ with μ outputs to the graph, where $\mu = 10(42m + 30n)$ (where, here and further, n is the number of variables in F and m is the number of clauses in F) so that we have sufficiently many outputs on hand to perform what follows. We connect some of the outputs and make them adjacent to the clause and variable vertices as follows (see Figure 3.3 for an illustration for clause vertices):

- For every clause vertex c_j:
  - We first add three new vertices e_j, f_j, g_j, joined via the edges e_jf_j, f_jg_j, and e_jg_j to form a triangle. We now identify e_j and the degree-1 vertex of each of four unused outputs of G_μ. Similarly, we identify f_j and the degree-1 vertex of each of four unused outputs of G_μ. We next identify g_j and the degree-1 vertex of each of two unused outputs of G_μ. We finally add the edge g_jc_j to the graph.
  - We then add three new vertices e'_j, f'_j, g'_j, forming a triangle. We then identify e'_j and five unused outputs of G_μ as above, f'_j and five unused outputs of G_μ, and g'_j and three unused outputs of G_μ. We finally add the edge g'_jc_j to the graph.

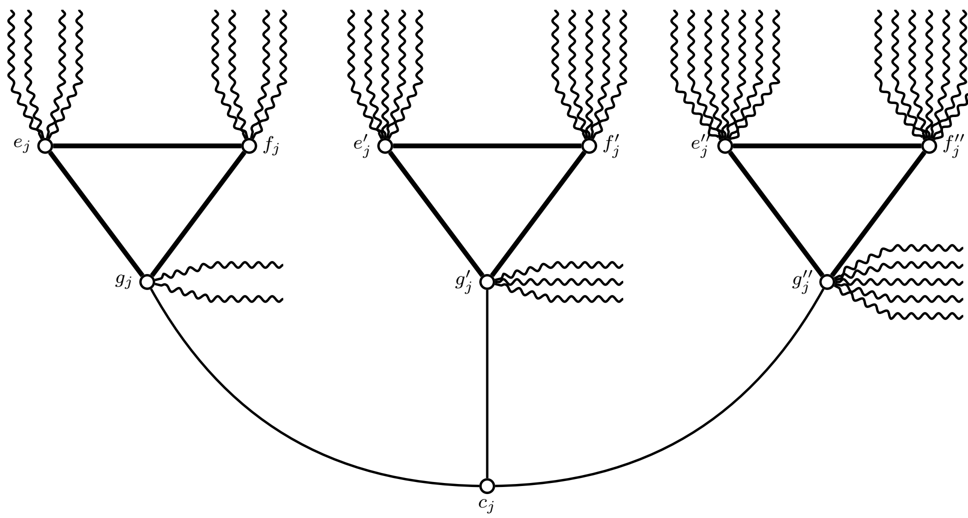


Figure 3.3 – Structure around a clause vertex c_j. Wiggly edges are outputs of G_μ.

  — We finally add three new vertices e''_j, f''_j, g''_j, forming a triangle. We then identify e''_j and seven unused outputs of G_μ as above, f''_j and seven unused outputs of G_μ, and g''_j and five unused outputs of G_μ. We finally add the edge $g''_j c_j$ to the graph.

— For every variable vertex v_i:
  — We first add three new vertices r_i, s_i, t_i joined to form a triangle (r_i, s_i, t_i, r_i). We then identify r_i and five unused outputs of G_μ as above. Similarly, we then identify s_i and five unused outputs of G_μ. Then, we identify t_i and three unused outputs of G_μ. Finally we add the edge $t_i v_i$ to the graph.
  — We then add three new vertices r'_i, s'_i, t'_i joined to form a triangle (r'_i, s'_i, t'_i, r'_i). We then identify r'_i and six unused outputs of G_μ as above. Similarly, we then identify s'_i and six unused outputs of G_μ. Then, we identify t'_i and four unused outputs of G_μ. Finally we add the edge $t'_i v_i$ to the graph.
  — Finally, we identify v_i with the degree-1 vertex of one unused output of G_μ.

Note that there are, at this point, a total of $\beta = 3m + 12m + 8n = 15m + 8n$ edges in the graph that are not part of G_μ. More precisely, $3m$ of these edges are edges of G_F, *i.e.*, formula edges, $12m$ of these edges are part or incident to the triangles added above and joined to the clause vertices, while $8n$ of these edges are part or incident to the triangles joined to the variable vertices. We refer to this graph, that is, the current one with β edges, and that contains none of the edges of G_μ, as G'_F.

Note that since $\mu = 10(42m + 30n)$, then, at this point, only $\frac{1}{10}\mu$ outputs of G_μ have been used. To each of the $\frac{9}{10}\mu$ unused outputs of G_μ, we plug a new copy of the corrector gadget C. Now, we add to the graph an initiator gadget I_α of length α that we plug to the input of G_μ, where α is chosen to be the unique integer such that

$$2 \leq ((6\alpha + 1) + (32\mu - 31)) - (4\alpha + (56\mu - 56) + \beta) \leq 3.$$

The whole resulting graph is our G, whose *input* is the input of I_α. Clearly, G is obtained in polynomial time from F.

Claim 3.1.5. *Let ℓ be a proper 2-labelling of G. If the input is assigned label 2, then ℓ cannot be equitable.*

Proof of the claim. Assume this is wrong, and consider ℓ an equitable proper 2-labelling of G assigning label 2 to the input. We investigate how many 1s and 2s must be assigned to several of the different gadgets that were plugged together to build G.

— Regarding the initiator gadget I_α of length α used in the construction of G, by Theorem 3.1.2 we get that ℓ must assign label 1 to exactly 4α edges and label 2 to exactly $6\alpha + 1$ edges.

— By Theorem 2.3.7, in G the input of each of the $\frac{9}{10}\mu$ copies of the corrector gadget C have their input labelled 2 by ℓ, since it coincides with an output of G_μ whose input is labelled 2 (by Theorem 3.1.2, since the output of I_α and the input of G_μ coincide). By Theorem 3.1.3, in each of these copies of C, there are two edges labelled 1 and seven edges labelled 2.

Omitting all of the contributions of the corrector gadgets, we can state that there are, at this point, at least $2\alpha + 1$ more 2s than 1s. This imbalance must be fixed via the labelling of the other edges of G_μ (*i.e.*, not the input of G_μ) and of G'_F. By Theorem 2.3.7, at most $56\mu - 56$ edges of G_μ can be assigned label 1 (which yield at least $32\mu - 31$ edges of G_μ labelled 2, due to the number of edges of G_μ), while the number of edges of G'_F is β. By our choice of α, it is then impossible that the number of assigned 1s by ℓ catches up with the number of assigned 2s. This contradicts the equitability of ℓ. ◇

Towards establishing the equivalence with a 1-in-3 truth assignment ϕ satisfying F, let us now see how a proper 2-labelling ℓ of G assigning label 1 to the input behaves. We start off by pointing out the following property of the triangles we have joined to the clause and variable vertices.

Claim 3.1.6. *Let $\gamma \geq 4$. Let H be any graph with a triangle (u, v, w, u) and an edge xw (where $x \notin \{u, v\}$), and ℓ be a partial proper 2-labelling of H. Assume only the edges uv, vw, wu, and xw remain to be labelled, that the partial colour of u and v is $\gamma - 2$, and that the partial colour of w is $\gamma - 4$. Then, in every proper extension of ℓ to uv, vw, wu, and xw, we have $\ell(uv) = 2$, $\ell(xw) = 1$, and $c_\ell(w) = \gamma$.*

Proof of the claim. Since only uv, vw, wu, and xw remain to be labelled, and u and v currently have the same partial colour, so that $c_\ell(u) \neq c_\ell(v)$ we have, say, $\ell(uw) = 1$ and $\ell(vw) = 2$. If $\ell(uv) = 1$, then we get $c_\ell(u) = \gamma$, $c_\ell(v) = \gamma+1$, while the partial colour of w is currently $\gamma-1$. It is then impossible to assign a correct label to xw, *i.e.*, so that $c_\ell(w) \notin \{\gamma, \gamma+1\} = \{c_\ell(u), c_\ell(v)\}$. So, we have $\ell(uv) = 2$, in which case $c_\ell(u) = \gamma + 1$ and $c_\ell(v) = \gamma + 2$. As above, the partial colour of w is currently $\gamma - 1$, and, so that $c_\ell(w) \notin \{\gamma+1, \gamma+2\}$, we must set $\ell(xw) = 1$. Then, $c_\ell(w) = \gamma$. ◇

Claim 3.1.6, applied to the structure of G (and more precisely to that of G'_F), yields the following.

Claim 3.1.7. *For any proper 2-labelling ℓ of G assigning label 1 to the input:*

— *For each clause vertex c_j, exactly one of its three incident formula edges is assigned label 1. Hence, $c_\ell(c_j) = 8$.*

— *For each variable vertex v_i, either all three of its incident formula edges are assigned label 1, or they are all assigned label 2. Hence, $c_\ell(v_i) \in \{6, 9\}$.*

— *The number of edges in G'_F that are assigned label 1 by ℓ is $7m + 4n$, while the number of edges assigned label 2 is $8m + 4n$.*

Proof of the claim. Let ℓ be such a labelling of G. By Theorems 3.1.2 and 2.3.7, all outputs of G_m must also be labelled 1 by ℓ.

— Consider any clause vertex c_j of G, and, in particular, the neighbouring triangle (e_j, f_j, g_j). Note that all the conditions are met to apply Claim 3.1.6. Similarly, this claim applies to the two triangles (e'_j, f'_j, g'_j) and (e''_j, f''_j, g''_j). From the claim, we get that $\ell(g_jc_j) = \ell(g'_jc_j) = \ell(g''_jc_j) = 1$, $c_\ell(g_j) = 6$, $c_\ell(g'_j) = 7$, and $c_\ell(g''_j) = 9$. Since c_j is incident to only three other edges, formula ones, one of them must be labelled 1 while the other two must be labelled 2 so that $c_\ell(c_j) \notin \{6, 7, 9\}$. Then, $c_\ell(c_j) = 8$.

— Consider any variable vertex v_i of G. By the same arguments, we have $\ell(t_iv_i) = \ell(t'_iv_i) = 1$, $c_\ell(t_i) = 7$, and $c_\ell(t'_i) = 8$. Consequently, the three remaining (formula) edges incident to v_i must either all be labelled 1 by ℓ, so that $c_\ell(v_i) = 6$, or all be labelled 2, so that $c_\ell(v_i) = 9$ (recall that v_i is also incident to an output of G_μ labelled 1). These are the only two possibilities so that $c_\ell(v_i) \notin \{7, 8\}$.

The last part of the statement follows from Claim 3.1.6 and the arguments above. This concludes the proof. Note, in particular, that a consequence is that we have $c_\ell(c_j) \neq c_\ell(v_i)$ for every clause vertex c_j and variable vertex v_i. $\diamond$

Claim 3.1.7 gives us a direct equivalence between finding a proper 2-labelling of G where the input is labelled 1 and a 1-in-3 truth assignment to the variables of F. Indeed, consider a proper 2-labelling ℓ of G assigning label 1 to the input. We regard the fact that $\ell(v_ic_j) = 1$ (respectively $\ell(v_ic_j) = 2$) as having, in F, variable v_i bringing truth value true (respectively false) to clause C_j. The condition in the first item of Claim 3.1.7 depicts the fact that, by a 1-in-3 truth assignment of F, a clause is considered satisfied only if it has exactly one true variable. The condition in the second item depicts the fact that, by a truth assignment, a variable brings the same truth value to all of the clauses that contain it. Thus, we can design a 1-in-3 truth assignment ϕ to the variables of F from ℓ, and *vice versa*.

Thus, F is 1-in-3 satisfiable if and only if G admits proper 2-labellings where the input is labelled 1. By Claim 3.1.5, all equitable proper 2-labellings of G (if any) must assign label 1 to the input. Thus, to prove that F is 1-in-3 satisfiable if and only if G admits equitable proper 2-labellings, it remains to show that G admits proper 2-labellings assigning label 1 to the input if and only if it admits equitable ones assigning label 1 to the input. Since every equitable proper labelling is a proper labelling, all that remains is to prove the following claim.

Claim 3.1.8. *If G admits proper 2-labellings where the input is assigned label 1, then G admits equitable proper 2-labellings where the input is assigned label 1.*

Proof of the claim. Let us consider a proper 2-labelling ℓ of G assigning label 1 to the input, obtained as follows. From Theorems 3.1.2 and 2.3.7, we know that all outputs of G_μ must also be assigned label 1. We propagate ℓ in I_α, G_μ, and G'_F while guaranteeing the following properties:

— In I_α, exactly 4α edges are assigned label 2 by ℓ, while $6\alpha + 1$ edges are assigned label 1. This is actually the only way to propagate ℓ in I_α, recall Theorem 3.1.2. Thus, here, there are $2\alpha + 1$ more assigned 1s than assigned 2s.

— In G_μ, the number of assigned 1s is as small as possible, *i.e.*, is $32\mu - 31$. In that case, the number of assigned 2s is $56\mu - 56$. Such a labelling can be achieved by Theorem 2.3.7.

— In G'_F, the number of assigned 2s is m more than the number of assigned 1s. By Claim 3.1.7 and since G admits a proper 2-labelling where the input is assigned label 1, this property is attainable (while maintaining that the labelling is proper) and actually has to hold.

To summarise the above, at this point, the number of assigned 1s is $((6\alpha+1)+(32\mu-31))-(4\alpha+(56\mu-56)+m)$ more than the number of assigned 2s. Recall that α was chosen as the unique integer such that $2 \leq ((6\alpha+1)+(32\mu-31))-(4\alpha+(56\mu-56)+\beta) \leq 3$. Thus, $\beta-m+2 \leq ((6\alpha+1)+(32\mu-31))-(4\alpha+(56\mu-56)+m) \leq \beta-m+3$, and hence, the number of assigned 1s we have considered is either $14m+8n+2$ or $14m+8n+3$ more than the number of assigned 2s (recall that $\beta = 15m+8n$). It remains to consider the $\frac{9}{10}\mu = 9(42m+30n)$ copies of the corrector gadget C in G. This means that the number of copies of C in G is much bigger than $14m+8n+3$. By Theorem 3.1.3, we can propagate ℓ to some copies of C so that six edges are assigned label 2 and three edges are assigned label 1. This way, the number of assigned 2s we have considered thus far catches up with the number of assigned 1s.

For the remaining copies of C, we can assume that ℓ roughly alternates propagating following the two labelling possibilities described in Theorem 3.1.3 when the input is labelled 1, so that the number of assigned 2s we have considered remains close yet slightly bigger than the number of assigned 1s. If, eventually, ℓ is not equitable because the number of assigned 2s is slightly bigger than the number of assigned 1s, then we can freely switch from 2 to 1 the labels assigned to some edges of, *e.g.*, triangles in the copies of T_2 in some of the spreading gadgets $G^{\curlywedge}$ in G_μ. Recall that all these edges are indeed currently assigned label 2 (since we have minimised the number of assigned 1s in G_μ).

Something to take into consideration is that the labelling of G_μ we have considered above, *i.e.*, the one minimising the number of 1s, does not comply with the two labelling schemes of the corrector gadget C. Indeed, when the spreading gadget $G^{\curlywedge}$ is labelled so that the input is labelled 1 and the number of 1s is minimised, note that the vertices u_9 and u_{12} must have colour 3, which is not compatible with the labelling of C in Figure 3.2(b). In this case, it is necessary to make u_9 (or u_{12}) have colour 2 (so that they comply with the desired labelling of C) by just changing to 1 the label of an incident edge labelled 2. This consequently makes the number of 2s increase, which must be taken into consideration for deciding how to label the next copies of C.

Eventually, ℓ is equitable. $\diamond$

To finish off the proof, we prove that, regardless of whether F is 1-in-3 satisfiable, there always exist proper 2-labellings of G. In other words, we always have $\chi_\Sigma(G) = 2$.

Claim 3.1.9. *There exist proper* 2*-labellings of* G*.*

Proof of the claim. We show that G admits proper 2-labellings ℓ where the input is labelled 2. Recall that we do not care about equitability here. By Theorem 3.1.2, the output of I_α, which is the input of G_μ, must be labelled 2 when its input is labelled 2. In turn, by Theorem 2.3.7, the outputs of G_μ must be labelled 2 as well. Some of these outputs are the input of corrector gadgets. From Theorems 3.1.2, 3.1.3, and 2.3.7, we get that there do exist partial proper 2-labellings ℓ of these gadgets in G such that no conflicts arise.

It remains to prove that such a partial labelling ℓ can properly be extended to the edges of G'_F. We demonstrate the arguments for one triangle (e_j, f_j, g_j, e_j) adjacent to a clause vertex c_j, but the arguments are identical regarding the other triangles and the variable vertices. Because e_j and f_j are incident to four edges of G_μ, which are all labelled 2 by the arguments above, e_j and f_j already have partial colour 8. Let us assign label 2 to e_jf_j and f_jg_j, and label 1 to e_jg_j. This gives $c_\ell(e_j) = 11 \neq c_\ell(f_j) = 12$. At this point, g_j has partial colour 7 (3 from the labelling of e_jg_j and f_jg_j, and 4 from the two outputs of G_μ). Let us assign label 1 to g_jc_j, so that $c_\ell(g_j) = 8$. Note that no two of e_j, f_j, and g_j are in conflict, and also their colour is so big that no conflict with adjacent vertices in G_μ can arise.

By repeating these arguments, for every clause vertex c_j, its incident edges c_jg_j, $c_jg'_j$, and $c_jg''_j$ are labelled 1, while g_j, g'_j, and g''_j have colour at least 8. In particular, c_j has partial colour 3 at this point. Similarly, for every variable vertex v_i its incident edges v_it_i and $v_it'_i$ are labelled 1, and t_i and t'_i have colour at least 10. Recall that an output of G_μ, which is labelled 2, is also attached to v_i. Then, the partial colour of v_i is 4 at this point. Let us finish the construction of the labelling ℓ by assigning label 1 to every formula edge. Since every clause vertex c_j and variable vertex v_i is incident to exactly three such edges, we get that $c_\ell(c_j) = 6$ and $c_\ell(v_i) = 7$ for every clause vertex c_j and every variable vertex v_i. Then, neither clause vertices nor variable vertices get in conflict with any of their neighbours. In particular, no clause vertex gets in conflict with a variable vertex. Then, ℓ is proper, as desired. $\diamond$

This concludes the proof. $\square$

Another interpretation of Theorem 3.1.4 is that, independently of whether Conjecture 2.1.2 (claiming that for almost all graphs G, $\chi_\Sigma(G) \leq 3$) is true or not, determining $\overline{\chi_\Sigma}(G)$ is an NP-hard problem for a given graph G. It is also worth mentioning that, in our reduction, the reduced graphs G we construct should always verify $\overline{\chi_\Sigma}(G) \leq 3$. This can be seen by noting that all gadgets and structures we have added to G themselves admit many equitable proper 3-labellings, some of which could possibly be combined to yield one of G.

3.2 Bipartite graphs G with $\chi_\Sigma(G) < \overline{\chi_\Sigma}(G)$

In this section, we investigate the existence of bipartite graphs G with $\chi_\Sigma(G) < \overline{\chi_\Sigma}(G)$. In Section 3.2.1, we first focus on bipartite graphs G with $\chi_\Sigma(G) = 3$, as they stand as good candidates of graphs that could have $\overline{\chi_\Sigma}(G) > 3$. We prove that, actually, $\overline{\chi_\Sigma}(G) = 3$ holds for all these graphs. In Section 3.2.2, we then study the existence of bipartite graphs G with $\chi_\Sigma(G) = 2$ and $\overline{\chi_\Sigma}(G) = 3$. We provide operations for building infinitely many such graphs.

3.2.1 Bipartite graphs G with $\chi_\Sigma(G) = 3$

Recall that a bipartite graph G verifying $\chi_\Sigma(G) = 3$ is an odd multi-cactus (as defined in Section 2.3.1). In this section we prove the following:

Theorem 3.2.1. *If G is an odd multi-cactus, then $\overline{\chi_\Sigma}(G) = 3$.*

Proof. The proof is by induction on the number of vertices of G. The base case corresponds to G being C_6, the cycle of length 6, which is the smallest odd multi-cactus. We first prove a more general case, namely that the claim is true whenever G is a cycle with length at least 6 congruent to 2 modulo 4.

Let G be a cycle with length at least 6 congruent to 2 modulo 4. In this case, an equitable proper 3-labelling can be obtained as follows. Traverse the successive edges of G starting from an arbitrary one, and assign labels $1, 2, 3, 1, 2, 3, \dots$ going along until all edges are labelled. Note that, doing so, at any moment of the procedure the resulting partial labelling is equitable.

— If the length of G is congruent to 0 or 1 modulo 3, then, by Observation 2.1.5, the resulting labelling is proper. This is because no two edges at distance 2 receive the same label, which is the only colour conflict that can occur in a path.

— If the length of G is congruent to 2 modulo 3 (the smallest such graph is C_{14}), then we get a conflict because of the last two edges that were labelled $l_1 = 1$ and $l_2 = 2$ respectively, which are each at distance 2 from an edge with the same label. In this situation, we change l_1 into 3 and l_2 into 1. Note that no conflict remains now. Furthermore, the labelling remains equitable (the number of assigned 2s is one less than the numbers of assigned 1s and 3s, which are equal).

For the general case: suppose that all odd multi-cacti with order at most some $x - 1$ admit an equitable proper 3-labelling, and let us consider odd multi-cacti with order x. If $x \not\equiv 2 \bmod 4$, then, by construction, there exist no such graphs on x vertices, and the claim is true. Thus, we assume that $x \equiv 2 \bmod 4$.

Let G be an odd multi-cactus with x vertices. Since G can be assumed to be different from a cycle, it was obtained from a cycle of length 2 modulo 4 by repeated path attachments onto olive edges. Due to the structure of G, it can be noted that there has to exist a olive edge uv where:

1. there exist $p \geq 1$ disjoint paths $P_1, \dots, P_p$ joining u and v, all of whose inner vertices have degree 2, and
2. the graph G' obtained by removing the inner vertices of the paths $P_1, \dots, P_p$ from G is an odd multi-cactus where u and v have degree 2.

First off, it can be assumed that all of the P_i's have length 5. This is a consequence of the following more general result:

Claim 3.2.2. *Let $P_9 = (v_1, \dots, v_9)$ be the path of length 8, and assume we are given a partial proper 3-labelling ℓ' of P_9 where only the four edges v_1v_2, v_2v_3, v_7v_8, and v_8v_9 are labelled, so that $\ell'(v_1v_2) \neq \ell'(v_7v_8)$ and $\ell'(v_2v_3) \neq \ell'(v_8v_9)$. Then, for any permutation $\{\alpha, \beta, \gamma\}$ of $\{1, 2, 3\}$, it is possible to extend ℓ' to a proper 3-labelling ℓ of P_9 where two of v_3v_4, v_4v_5, v_5v_6, and v_6v_7 are labelled α, one of these edges is labelled β, and one of these edges is labelled γ.*

Proof of the claim. So that a labelling of P_9 is proper, we must only ensure that every two edges at distance 2 receive distinct labels, recall Observation 2.1.5. In particular, this implies that labelling v_3v_4 and v_5v_6 can be done independently from labelling v_4v_5 and v_6v_7. Then, we will be done if we can prove that labels α and β can correctly be assigned to v_3v_4 and v_5v_6, while labels α and γ can correctly be assigned to v_4v_5 and v_6v_7.

Without loss of generality, let us assume we want to assign labels α and β to v_3v_4 and v_5v_6. Note that, at this point, $\ell(v_3v_4)$ must only differ from $\ell(v_1v_2)$. Let us assume that we can assign $\ell(v_3v_4) = \alpha$ without there being a conflict with $\ell(v_1v_2)$, *i.e.*, $\ell(v_1v_2) \neq \alpha$. If no conflict arises upon setting $\ell(v_5v_6) = \beta$, then we are done. Otherwise, it means $\ell(v_7v_8) = \beta$. In that situation, let us instead set $\ell(v_5v_6) = \alpha$ and $\ell(v_3v_4) = \beta$. If this raises a conflict, this must be because $\ell(v_1v_2) = \beta$. But then, we deduce that $\ell(v_1v_2) = \ell(v_7v_8) = \beta$, a contradiction. $\diamond$

Indeed, assume, without loss of generality, that P_1 has length $4k + 1$ for some $k \geq 2$. Let us denote by $(u, v_1, \dots, v_{4k}, v)$ the successive vertices of P_1 from u to v. Let G' be the graph obtained from G by removing the edges $v_2v_3, v_3v_4, v_4v_5, v_5v_6, v_6v_7$ and joining the vertices v_2 and v_7 by an edge e. Note that G' is an odd multi-cactus since we have essentially contracted a path of length $4k + 1$ (with $k \geq 2$) into a path of length $4(k - 1) + 1$. Then, by the induction hypothesis, there is an equitable proper 3-labelling ℓ' of G'. By definition, note that $\ell'(uv_1) \neq \ell'(e)$, $\ell'(e) \neq \ell'(v_8v_9)$ (or $\ell'(e) \neq \ell'(v_8v)$ if $k = 2$), and $\ell'(v_1v_2) \neq \ell'(v_7v_8)$. To extend ℓ' to an equitable proper 3-labelling ℓ of G, for every edge that is both in G and G' we first infer the label by ℓ' to ℓ. We then set $\ell(v_2v_3)$ to $\ell'(e)$. Note that no conflict arises in G by this partial labelling. Furthermore, since ℓ' is equitable, so is ℓ. Now, consider $\{\alpha, \beta, \gamma\}$ a permutation of $\{1, 2, 3\}$ such that every two of $\mathrm{nb}_{\ell'}(\alpha) + 2$, $\mathrm{nb}_{\ell'}(\beta) + 1$, and $\mathrm{nb}_{\ell'}(\gamma) + 1$ differ by at most 1. Such an $\{\alpha, \beta, \gamma\}$ exists since ℓ' is equitable. By Claim 3.2.2, the current ℓ can be extended to the edges $v_3v_4, v_4v_5, v_5v_6, v_6v_7$ so that a proper 3-labelling of G results, and this can be done by assigning label α twice, and each of β and γ once. By our choice of α, β, γ, such a resulting labelling is also equitable.

Then, we can assume all P_i's have length exactly 5. For every $i \in \{1, \ldots, p\}$, let us set $P_i = (u, v_1^i, \ldots, v_4^i, v)$. Let us also denote by u' the second neighbour (different from v) of u in G', and by v' the second neighbour (different from u) of v in G'. Our goal is to extend ℓ' to the edges of the P_i's so that no conflict arises, and the resulting 3-labelling ℓ of G is proper. To begin, consider $\{\alpha, \beta, \gamma\}$ a permutation of $\{1, 2, 3\}$. The choice of α, β, and γ can be done in such a way that the ensuing labelling ℓ is equitable. Precisely, if $\mathrm{nb}_{\ell'}(1) = \mathrm{nb}_{\ell'}(2) = \mathrm{nb}_{\ell'}(3)$ or $\mathrm{nb}_{\ell'}(1) + 1 = \mathrm{nb}_{\ell'}(2) = \mathrm{nb}_{\ell'}(3)$ or $\mathrm{nb}_{\ell'}(1) = \mathrm{nb}_{\ell'}(2) = \mathrm{nb}_{\ell'}(3) - 1$, then $(\alpha, \beta, \gamma) = (1, 2, 3)$, else if $\mathrm{nb}_{\ell'}(1) - 1 = \mathrm{nb}_{\ell'}(2) = \mathrm{nb}_{\ell'}(3)$ or $\mathrm{nb}_{\ell'}(1) = \mathrm{nb}_{\ell'}(2) + 1 = \mathrm{nb}_{\ell'}(3)$, then $(\alpha, \beta, \gamma) = (2, 3, 1)$, and if $\mathrm{nb}_{\ell'}(1) = \mathrm{nb}_{\ell'}(2) = \mathrm{nb}_{\ell'}(3) + 1$ or $\mathrm{nb}_{\ell'}(1) = \mathrm{nb}_{\ell'}(2) - 1 = \mathrm{nb}_{\ell'}(3)$, then $(\alpha, \beta, \gamma) = (3, 1, 2)$. For all $1 \le i \le p$ and for any $a, b \in P_i$ such that $ab \in E(G)$, it is easy to verify that $c_\ell(a) \ne c_\ell(b)$, for any of the labellings ℓ proposed below.

In what follows, let x be any vertex in $X = N(u) \bigcap\limits_{i=1}^{p} P_i$, let y be any vertex in $Y = N(v) \bigcap\limits_{i=1}^{p} P_i$, and let w be any vertex in $W = (N(u) \cup N(v)) \bigcap\limits_{i=1}^{p} P_i$. Also, for all $1 \le i \le p$, let $\ell(P_i) = (\ell(uv_1^i), \ell(v_1^iv_2^i), \ell(v_2^iv_3^i), \ell(v_3^iv_4^i), \ell(v_4^iv))$.

Case $p = 2$:

All the possible subcases are illustrated in Table 3.1. Note that, in all of these subcases, the labelling ℓ has the property that $\sum\limits_{x \in X} \ell(ux) = \sum\limits_{y \in Y} \ell(vy)$, and so, $c_\ell(u) \ne c_\ell(v)$ since $c_{\ell'}(u) \ne c_{\ell'}(v)$ by the inductive hypothesis. Furthermore, note that the maximum colour of any vertex $w \in W$ is 6, no matter the labelling. Lastly, it may seem that the subcase $c_{\ell'}(u) + \alpha + \beta \ne c_{\ell'}(u')$, $c_{\ell'}(v) + \alpha + \beta = c_{\ell'}(v')$ has not been treated, but it is actually symmetric to the subcase where $c_{\ell'}(u) + \alpha + \beta = c_{\ell'}(u')$, $c_{\ell'}(v) + \alpha + \beta \ne c_{\ell'}(v')$, which has been treated through subcases 2-4 in Table 3.1.

Case $p \ge 3$:

Give the same labellings as in the case $p = 2$ for P_1 and P_2. For the remainder of the paths, simply label them so that the labelling ℓ is equitable (and proper) and so that, for all $3 \le j \le p$, we have $\ell(uv_1^j) = \ell(vv_4^j)$. Note that, in this case, for all $x \in X$, it is not possible for $c_\ell(u) = c_\ell(x)$. Indeed, if $c_\ell(u) \ge 7$, then we are done since, for all $w \in W$, we have that $c_\ell(w) \le 6$. Otherwise, if $c_\ell(u) = 6$, then, for all $x \in X$, we have that $\ell(ux) \le 2$, and so, $c_\ell(x) \le 5$. Lastly, if $c_\ell(u) = 5$ (note that this is the last case since $c_\ell(u) \ge 5$), then, for all $x \in X$, we have that $\ell(ux) = 1$, and so, $c_\ell(x) \le 4$. The same can be said for all $y \in Y$ and v.

Case $p = 1$:

All the possible subcases are illustrated in Table 3.2. Note that, in the first six of these subcases, the labelling ℓ has the property that $\ell(uv_1) = \ell(vv_4)$, and so, $c_\ell(u) \ne c_\ell(v)$ since $c_{\ell'}(u) \ne c_{\ell'}(v)$ by the inductive hypothesis. Furthermore, note that the logical subcase that would follow the last subcase in Table 3.2 would be that $c_{\ell'}(u) + \alpha = c_{\ell'}(u')$, $c_{\ell'}(v) + \beta = c_{\ell'}(v')$, $c_{\ell'}(u) + \beta \ne$

Subcase conditions	**Labelling of paths P_1 and P_2**	**No conflicts between u (v resp.) and any of its neighbours in P_1 or P_2**
$c_{\ell'}(u)+\alpha+\beta \neq c_{\ell'}(u')$, $c_{\ell'}(v)+\alpha+\beta \neq c_{\ell'}(v')$	$\ell(P_1)=(\alpha,\beta,\gamma,\alpha,\beta)$ $\ell(P_2)=(\beta,\alpha,\gamma,\gamma,\alpha)$	If $\alpha=1$: $c_\ell(w)\leq 4$ & $c_\ell(u),c_\ell(v)\geq 5$. If $\alpha=2$: $c_\ell(u),c_\ell(v)\geq 7$. If $\alpha=3$: $c_\ell(w)\leq 5$ & $c_\ell(u),c_\ell(v)\geq 6$.
$c_{\ell'}(u)+\alpha+\beta = c_{\ell'}(u')$, $c_{\ell'}(v)+2\alpha \neq c_{\ell'}(v')$, $c_{\ell'}(v)+2\alpha \neq \alpha+\gamma$	$\ell(P_1)=(\alpha,\beta,\beta,\gamma,\alpha)$ $\ell(P_2)=(\alpha,\beta,\gamma,\gamma,\alpha)$	If $\alpha=1$: $c_\ell(x)=3$ & $c_\ell(u)\geq 4$. If $\alpha=2$: $c_\ell(x)=5$ & $c_\ell(u)\geq 6$. If $\alpha=3$: $c_\ell(u)\geq 8$.
$c_{\ell'}(u)+\alpha+\beta = c_{\ell'}(u')$, $c_{\ell'}(v)+2\alpha \neq c_{\ell'}(v')$, $c_{\ell'}(v)+2\alpha = \alpha+\gamma$	$\ell(P_1)=(\alpha,\gamma,\beta,\beta,\alpha)$ $\ell(P_2)=(\alpha,\gamma,\gamma,\beta,\alpha)$	Since $c_{\ell'}(u)\neq c_{\ell'}(v)$, then $c_{\ell'}(u)+2\alpha\neq\alpha+\gamma$.
$c_{\ell'}(u)+\alpha+\beta = c_{\ell'}(u')$, $c_{\ell'}(v)+2\alpha = c_{\ell'}(v')$	$\ell(P_1)=(\gamma,\alpha,\beta,\gamma,\alpha)$ $\ell(P_2)=(\alpha,\gamma,\beta,\beta,\gamma)$	If $\alpha=1$: $c_\ell(w)\leq 5$ & $c_\ell(u),c_\ell(v)\geq 6$. If $\alpha=2$: $c_\ell(w)\leq 4$ & $c_\ell(u),c_\ell(v)\geq 5$. If $\alpha=3$: $c_\ell(u),c_\ell(v)\geq 7$.

Table 3.1 – The four subcases of the proof of Theorem 3.2.1 for $p=2$.

$c_{\ell'}(v)+\alpha$, $c_{\ell'}(v)+\alpha=2\alpha$, and $c_{\ell'}(u)+\beta=\beta+\alpha$. However, this subcase cannot exist since, if $c_{\ell'}(v)+\alpha=2\alpha$, then $c_{\ell'}(v)=\alpha$, and if $c_{\ell'}(u)+\beta=\beta+\alpha$, then $c_{\ell'}(u)=\alpha$, and hence, we have that $c_{\ell'}(u)=c_{\ell'}(v)$, a contradiction. Lastly, it may seem that the subcase $c_{\ell'}(u)+\alpha\neq c_{\ell'}(u')$, $c_{\ell'}(v)+\alpha=c_{\ell'}(v')$ has not been treated, but again, it is actually symmetric to the subcase where $c_{\ell'}(u)+\alpha=c_{\ell'}(u')$, $c_{\ell'}(v)+\alpha\neq c_{\ell'}(v')$, which has been treated through subcases 4-12 in Table 3.2. This concludes the proof as all of the possible cases have now been covered. □

3.2.2 Bipartite graphs G with $\chi_\Sigma(G)=2$

We start by introducing two operations, Operations 1 and 2, which, when applied to graphs G with $\overline{\chi_\Sigma}(G)\geq 3$, provide more graphs G' with $\overline{\chi_\Sigma}(G')\geq 3$.

Proposition 3.2.3 (Operation 1)**.** *Let G be a multigraph with $\overline{\chi_\Sigma}(G)\geq 3$. If G has an edge uv with multiplicity at least 2, then the graph G' obtained from G by subdividing one of these edges uv four times verifies $\overline{\chi_\Sigma}(G')\geq 3$.*

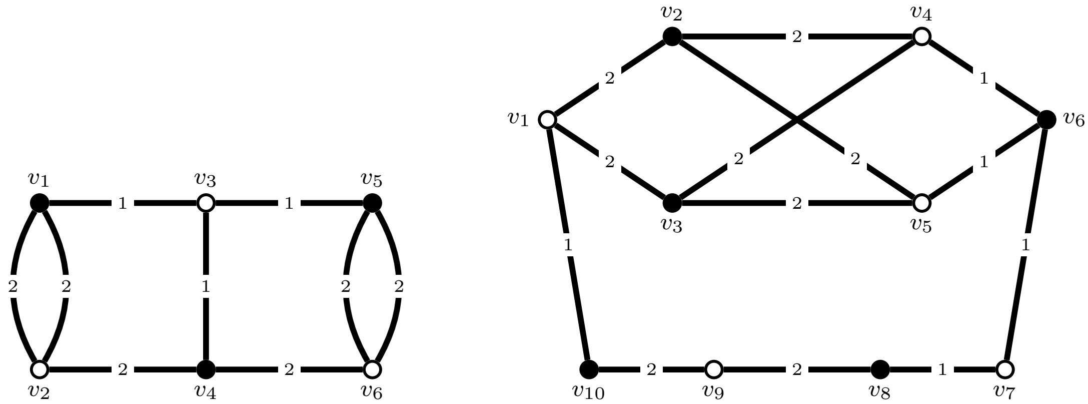
Proof. Let G' be obtained from G by replacing one edge uv with a path (u,w,x,y,z,v) of length 5. Assume there exists ℓ', an equitable proper 2-labelling of G'. Assume that $\ell'(uw)=\alpha$ for some $\alpha\in\{1,2\}$, and that $\ell'(wx)=\beta$ for some $\beta\in\{1,2\}$. Set $\{\overline{\alpha}\}=\{1,2\}\setminus\{\alpha\}$ and $\{\overline{\beta}\}=\{1,2\}\setminus\{\beta\}$. Then, by Observation 2.1.5, we have $\ell'(xy)=\overline{\alpha}$, $\ell'(yz)=\overline{\beta}$, and $\ell'(zv)=\alpha$. By the properness of ℓ', since u and v are adjacent in G', we have that $c_{\ell'}(u)\neq c_{\ell'}(v)$. This implies that the 2-labelling ℓ of G obtained from ℓ' by assigning label α to the edge uv that was subdivided for constructing G', and setting $\ell(e)=\ell'(e)$ for every $e\in E(G)\cap E(G')$, is proper. Furthermore, we have $\{\ell'(wx),\ell'(yz)\}=\{\ell'(xy),\ell'(zv)\}=\{1,2\}$. Hence, $\mathrm{nb}_\ell(1)=\mathrm{nb}_{\ell'}(1)-2$ and $\mathrm{nb}_\ell(2)=\mathrm{nb}_{\ell'}(2)-2$. So, ℓ is an equitable proper 2-labelling of G, a contradiction. Thus, $\overline{\chi_\Sigma}(G')\geq 3$. □

Proposition 3.2.4 (Operation 2)**.** *Let G be a graph with $\overline{\chi_\Sigma}(G)\geq 3$. If G has an edge uv with $d(u)=d(v)=2$, then the graph G' obtained from G by subdividing uv four times verifies $\overline{\chi_\Sigma}(G')\geq 3$.*

Subcase conditions	Labelling of path P_1	No conflicts between u (v resp.) and any of its neighbours in P_1
$c_{\ell'}(u)+\alpha \neq c_{\ell'}(u')$, $c_{\ell'}(v)+\alpha \neq c_{\ell'}(v')$, $c_{\ell'}(v)+\alpha \neq \beta+\alpha$, $c_{\ell'}(u)+\alpha \neq \gamma+\alpha$	$\ell(P_1)=(\alpha,\gamma,\beta,\beta,\alpha)$	By the conditions of the subcase.
$c_{\ell'}(u)+\alpha \neq c_{\ell'}(u')$, $c_{\ell'}(v)+\alpha \neq c_{\ell'}(v')$, $c_{\ell'}(v)+\alpha \neq \beta+\alpha$, $c_{\ell'}(u)+\alpha = \gamma+\alpha$	$\ell(P_1)=(\alpha,\beta,\beta,\gamma,\alpha)$	Since $c_{\ell'}(u) \neq c_{\ell'}(v)$, then $c_{\ell'}(v)+\alpha \neq \gamma+\alpha$.
$c_{\ell'}(u)+\alpha \neq c_{\ell'}(u')$, $c_{\ell'}(v)+\alpha \neq c_{\ell'}(v')$, $c_{\ell'}(v)+\alpha = \beta+\alpha$	$\ell(P_1)=(\alpha,\beta,\beta,\gamma,\alpha)$	Since $c_{\ell'}(u) \neq c_{\ell'}(v)$, then $c_{\ell'}(u)+\alpha \neq \beta+\alpha$.
$c_{\ell'}(u)+\alpha = c_{\ell'}(u')$, $c_{\ell'}(v)+\beta \neq c_{\ell'}(v')$, $c_{\ell'}(v)+\beta \neq \beta+\gamma$, $c_{\ell'}(u)+\beta \neq \beta+\alpha$	$\ell(P_1)=(\beta,\alpha,\alpha,\gamma,\beta)$	By the conditions of the subcase.
$c_{\ell'}(u)+\alpha = c_{\ell'}(u')$, $c_{\ell'}(v)+\beta \neq c_{\ell'}(v')$, $c_{\ell'}(v)+\beta \neq \beta+\gamma$, $c_{\ell'}(u)+\beta = \beta+\alpha$	$\ell(P_1)=(\beta,\gamma,\alpha,\alpha,\beta)$	Since $c_{\ell'}(u) \neq c_{\ell'}(v)$, then $c_{\ell'}(v)+\beta \neq \beta+\alpha$.
$c_{\ell'}(u)+\alpha = c_{\ell'}(u')$, $c_{\ell'}(v)+\beta \neq c_{\ell'}(v')$, $c_{\ell'}(v)+\beta = \beta+\gamma$	$\ell(P_1)=(\beta,\gamma,\alpha,\alpha,\beta)$	Since $c_{\ell'}(u) \neq c_{\ell'}(v)$, then $c_{\ell'}(u)+\beta \neq \beta+\gamma$.
$c_{\ell'}(u)+\alpha = c_{\ell'}(u')$, $c_{\ell'}(v)+\beta = c_{\ell'}(v')$, $c_{\ell'}(u)+\beta = c_{\ell'}(v)+\alpha$, $c_{\ell'}(v)+\gamma \neq \gamma+\alpha$, $c_{\ell'}(u)+\beta \neq 2\beta$	$\ell(P_1)=(\beta,\beta,\alpha,\alpha,\gamma)$	By the conditions of the subcase. Note also that $c_\ell(u) \neq c_\ell(v)$.
$c_{\ell'}(u)+\alpha = c_{\ell'}(u')$, $c_{\ell'}(v)+\beta = c_{\ell'}(v')$, $c_{\ell'}(u)+\beta = c_{\ell'}(v)+\alpha$, $c_{\ell'}(v)+\gamma \neq \gamma+\alpha$, $c_{\ell'}(u)+\beta = 2\beta$	$\ell(P_1)=(\gamma,\alpha,\beta,\beta,\alpha)$	Since $c_{\ell'}(u)=\beta$, then $c_{\ell'}(v)=2\beta-\alpha$, $c_{\ell'}(u')=\beta+\alpha$, $c_{\ell'}(v')=3\beta-\alpha$, $c_\ell(u)=\beta+\gamma$, and $c_\ell(v)=2\beta$. It can then be verified that there are no conflicts.
$c_{\ell'}(u)+\alpha = c_{\ell'}(u')$, $c_{\ell'}(v)+\beta = c_{\ell'}(v')$, $c_{\ell'}(u)+\beta = c_{\ell'}(v)+\alpha$, $c_{\ell'}(v)+\gamma = \gamma+\alpha$	$\ell(P_1)=(\beta,\alpha,\alpha,\beta,\gamma)$	By the last two conditions of the subcase, $c_{\ell'}(u)+\beta=2\alpha$. Thus, $c_\ell(u) \neq \beta+\alpha$. Note also that $c_\ell(u) \neq c_\ell(v)$.
$c_{\ell'}(u)+\alpha = c_{\ell'}(u')$, $c_{\ell'}(v)+\beta = c_{\ell'}(v')$, $c_{\ell'}(u)+\beta \neq c_{\ell'}(v)+\alpha$, $c_{\ell'}(v)+\alpha \neq 2\alpha$, $c_{\ell'}(u)+\beta \neq 2\beta$	$\ell(P_1)=(\beta,\beta,\gamma,\alpha,\alpha)$	By the conditions of the subcase. Note also that $c_\ell(u) \neq c_\ell(v)$.
$c_{\ell'}(u)+\alpha = c_{\ell'}(u')$, $c_{\ell'}(v)+\beta = c_{\ell'}(v')$, $c_{\ell'}(u)+\beta \neq c_{\ell'}(v)+\alpha$, $c_{\ell'}(v)+\alpha \neq 2\alpha$, $c_{\ell'}(u)+\beta = 2\beta$	$\ell(P_1)=(\beta,\alpha,\gamma,\beta,\alpha)$	Note that $\beta \geq 2$ since $c_{\ell'}(u)=\beta$. If $\beta=2$: $c_\ell(u)=4$, $c_\ell(v)=5$, $c_\ell(v')=6$ & $c_\ell(u')=c_\ell(x)=c_\ell(y)=3$. If $\beta=3$: $c_\ell(u)=6$, $c_\ell(v)=4$, $c_\ell(v')=7$ & $c_\ell(u')=c_\ell(x)=c_\ell(y)=5$.
$c_{\ell'}(u)+\alpha = c_{\ell'}(u')$, $c_{\ell'}(v)+\beta = c_{\ell'}(v')$, $c_{\ell'}(u)+\beta \neq c_{\ell'}(v)+\alpha$, $c_{\ell'}(v)+\alpha = 2\alpha$, $c_{\ell'}(u)+\beta \neq \beta+\alpha$	$\ell(P_1)=(\beta,\alpha,\gamma,\beta,\alpha)$	By the conditions of the subcase. Note also that $c_\ell(u) \neq c_\ell(v)$.

Table 3.2 – The twelve subcases of the proof of Theorem 3.2.1 for $p=1$.

Proof. Let us denote by (u,w,x,y,z,v) the path of length 5 of G' that results from the subdivision of uv. Also, let u' and v' be the other neighbours of u and v respectively. Assume there exists ℓ', an equitable proper 2-labelling of G' such that $\ell'(u'u)=\alpha$ for some $\alpha \in \{1,2\}$, and that $\ell'(uw)=\beta$ for some $\beta \in \{1,2\}$. Set $\{\overline{\alpha}\}=\{1,2\}\setminus\{\alpha\}$ and $\{\overline{\beta}\}=\{1,2\}\setminus\{\beta\}$. According to Observation 2.1.5, we have $\ell'(wx)=\overline{\alpha}$, $\ell'(xy)=\overline{\beta}$, $\ell'(yz)=\alpha$, $\ell'(zv)=\beta$, and $\ell'(vv')=\overline{\alpha}$.



(a) With parallel edges. (b) With adjacent degree-2 vertices.

Figure 3.4 – Proper 2-labellings of two bipartite graphs G with $2 = \chi_\Sigma(G) < \overline{\chi_\Sigma}(G) = 3$.

It follows that the 2-labelling ℓ of G obtained from ℓ' by setting $\ell(uv) = \beta$ and $\ell(e) = \ell'(e)$ for every $e \in E(G) \cap E(G')$ is proper. In particular, note that the above implies that $c_\ell(u) = c_{\ell'}(u) \neq c_{\ell'}(v) = c_\ell(v)$, although u and v are not adjacent in G'. Furthermore, we have $\{\ell'(wx), \ell'(yz)\} = \{\ell'(xy), \ell'(zv)\} = \{1, 2\}$. This implies that $\mathrm{nb}_\ell(1) = \mathrm{nb}_{\ell'}(1) - 2$ and $\mathrm{nb}_\ell(2) = \mathrm{nb}_{\ell'}(2) - 2$. So, ℓ is an equitable proper 2-labelling of G, a contradiction. Thus, $\overline{\chi_\Sigma}(G') \geq 3$. □

We note in particular that Operations 1 and 2 mentioned in Propositions 3.2.3 and 3.2.4, when performed on bipartite graphs, yield graphs that are also bipartite. Also, by carefully applying these operations, we can make sure that the resulting graphs are not odd multi-cacti. From this observation, we come up with two infinite families of bipartite graphs G verifying $2 = \chi_\Sigma(G) < \overline{\chi_\Sigma}(G) = 3$.

The first such family is obtained by repeatedly applying Operations 1 and 2 from the cubic bipartite multigraph depicted in Figure 3.4(a). This graph indeed has the following properties:

Proposition 3.2.5. *Let G be the cubic bipartite multigraph depicted in Figure 3.4(a). Then, $\chi_\Sigma(G) = 2$ and $\overline{\chi_\Sigma}(G) = 3$.*

Proof. Since G is cubic, we have $\chi_\Sigma(G) > 1$ and thus, $\overline{\chi_\Sigma}(G) > 1$. Actually, we even have $\chi_\Sigma(G) = 2$ since G does not match the definition of an odd multi-cactus (a proper 2-labelling is also included in Figure 3.4(a)). Towards a contradiction, assume G admits an equitable proper 2-labelling ℓ. In order to have $c_\ell(v_1) \neq c_\ell(v_2)$, we have $\ell(v_1v_3) \neq \ell(v_2v_4)$. Similarly, since $c_\ell(v_5) \neq c_\ell(v_6)$, we have $\ell(v_5v_3) \neq \ell(v_6v_4)$. Now, since $c_\ell(v_3) \neq c_\ell(v_4)$, we have $\ell(v_1v_3) = \ell(v_3v_5)$ and $\ell(v_2v_4) = \ell(v_4v_6)$, since otherwise, $c_\ell(v_3) = c_\ell(v_4) = 1 + 2 + \ell(v_3v_4)$ by the previous argument. Thus, without loss of generality, we may assume that $\ell(v_1v_3) = \ell(v_3v_5) = 1$ and $\ell(v_2v_4) = \ell(v_4v_6) = 2$.

Assume now that $\ell(v_3v_4) = 1$. This gives $c_\ell(v_3) = 3$ and $c_\ell(v_4) = 5$. Now, note that the two edges joining v_1 and v_2, and similarly the two edges joining v_5 and v_6, cannot both be assigned label 1 (as, otherwise, v_1 or v_5 would be in conflict with v_3). Similarly, to avoid a conflict with v_4, the two edges joining v_1 and v_2, and similarly the two edges joining v_5 and v_6, cannot be assigned labels 1 and 2 respectively. Thus, these four edges must be assigned label 2, which means that $\mathrm{nb}_\ell(1) = 3$ and $\mathrm{nb}_\ell(2) = 6$. This is a contradiction to the equitability of ℓ. Similar arguments can be used to show that we cannot have $\ell(v_3v_4) = 2$ either. Thus, $\overline{\chi_\Sigma}(G) > 2$, and one can easily come up with equitable proper 3-labellings of G. □

A second infinite family of bipartite graphs G with $2 = \chi_\Sigma(G) < \overline{\chi_\Sigma}(G) = 3$ is obtained by repeatedly applying Operation 2, described in Theorem 3.2.4, to the graph depicted in Figure 3.4(b) (note that this is $K_{3,3}$ with one edge subdivided four times). Note that this does not result in an odd multi-cactus. The graph depicted in Figure 3.4(b) has the following properties:

Proposition 3.2.6. *Let G be the subcubic bipartite graph depicted in Figure 3.4(b). Then, $\chi_\Sigma(G) = 2$ and $\overline{\chi_\Sigma}(G) = 3$.*

Proof. Again, G is not locally irregular and does not match the structure of an odd multi-cactus, so $\chi_\Sigma(G) = 2$ (a proper 2-labelling is also included in Figure 3.4(b)). We now prove that $\overline{\chi_\Sigma}(G) = 3$. Let us suppose that there exists an equitable proper 2-labelling ℓ of G. By Observation 2.1.5, we must have $\ell(v_1v_{10}) \neq \ell(v_9v_8)$ and $\ell(v_7v_6) \neq \ell(v_9v_8)$, and thus, $\ell(v_1v_{10}) = \ell(v_7v_6)$. Moreover, either $\ell(v_{10}v_9) = \ell(v_1v_{10}) = \ell(v_7v_6)$ and $\ell(v_9v_8) = \ell(v_8v_7)$ or $\ell(v_1v_{10}) = \ell(v_8v_7) = \ell(v_7v_6)$ and $\ell(v_{10}v_9) = \ell(v_9v_8)$. This implies that each of the labels 1 and 2 appears exactly twice in the edges $v_{10}v_9$, v_9v_8, v_8v_7, v_7v_6.

Let G' be the graph obtained from G by replacing the path $P = (v_1, v_{10}, v_9, v_8, v_7, v_6)$ by a single edge v_1v_6. Moreover, let ℓ' be the labelling of G' such that $\ell'(e) = \ell(e)$ for every edge $e \in E(G') \cap E(G)$, and $\ell'(v_1v_6) = \ell(v_1v_{10})$. Since ℓ is equitable and due to the previous remark, ℓ' is an equitable 2-labelling of G. Now, it suffices to show that ℓ' is also proper. If this is the case, we arrive at a contradiction since G' is isomorphic to $K_{3,3}$ and $\overline{\chi_\Sigma}(K_{3,3}) = 3$ (as proved in [18]).

For the sake of contradiction, suppose that ℓ' is not proper. Since ℓ is a proper labelling of G, it follows that $c_{\ell'}(v_1) = c_{\ell'}(v_6)$ in G' and that these are the only two vertices that are in conflict. Observe that G' is a cubic graph and thus, for each $v \in V(G')$, we have $c_{\ell'}(v) \in \{3, 4, 5, 6\}$. We distinguish the following cases:

— $c_{\ell'}(v_1) = c_{\ell'}(v_6) = 3$.
In this case, $\ell'(v_4v_6) = \ell'(v_2v_1) = \ell'(v_3v_1) = \ell'(v_5v_6) = 1$, and so, $c_{\ell'}(v_2), c_{\ell'}(v_3), c_{\ell'}(v_4), c_{\ell'}(v_5) \neq 6$. Moreover, so that $3 = c_{\ell'}(v_1) \notin \{c_{\ell'}(v_2), c_{\ell'}(v_3)\}$ and $3 = c_{\ell'}(v_6) \notin \{c_{\ell'}(v_4), c_{\ell'}(v_5)\}$, we have that $c_{\ell'}(v_2), c_{\ell'}(v_3), c_{\ell'}(v_4), c_{\ell'}(v_5) \in \{4, 5\}$. By symmetry, let us assume that $c_{\ell'}(v_2) = 5$. Then, we must have $c_{\ell'}(v_4) = c_{\ell'}(v_5) = 4$ which means $\ell'(v_3v_4) = 1$ and that we must have $c_{\ell'}(v_3) = 5$, which is impossible since $\ell'(v_3v_4) = 1$.

— $c_{\ell'}(v_1) = c_{\ell'}(v_6) = 4$.
First, let us assume that $\ell'(v_1v_6) = 2$. In this case, $\ell'(v_4v_6) = \ell'(v_2v_1) = \ell'(v_3v_1) = \ell'(v_5v_6) = 1$, and so, $c_{\ell'}(v_2), c_{\ell'}(v_3), c_{\ell'}(v_4), c_{\ell'}(v_5) \neq 6$. Moreover, so that $4 = c_{\ell'}(v_1) \notin \{c_{\ell'}(v_2), c_{\ell'}(v_3)\}$ and $4 = c_{\ell'}(v_6) \notin \{c_{\ell'}(v_4), c_{\ell'}(v_5)\}$, we have that $c_{\ell'}(v_2), c_{\ell'}(v_3), c_{\ell'}(v_4), c_{\ell'}(v_5) \in \{3, 5\}$. By symmetry, let us assume that $c_{\ell'}(v_2) = 5$. Then, we have $c_{\ell'}(v_4) \in \{4, 5\}$, which conflicts with either v_2 or v_6.
Second, let us assume that $\ell'(v_1v_6) = 1$. In this case, we may assume by symmetry that $\ell'(v_4v_6) = \ell'(v_2v_1) = 1$ and $\ell'(v_3v_1) = \ell'(v_5v_6) = 2$, and so, $c_{\ell'}(v_2), c_{\ell'}(v_4) \in \{3, 5\}$. By symmetry, let us assume that $c_{\ell'}(v_2) = 5$. Therefore, $\ell'(v_2v_4) = 2$ and $c_{\ell'}(v_4) \in \{4, 5\}$ which conflicts either with v_2 or v_6.

— $c_{\ell'}(v_1) = c_{\ell'}(v_6) \in \{5, 6\}$.
These cases can be proved similarly to the previous cases by switching labels 1 and 2.

□

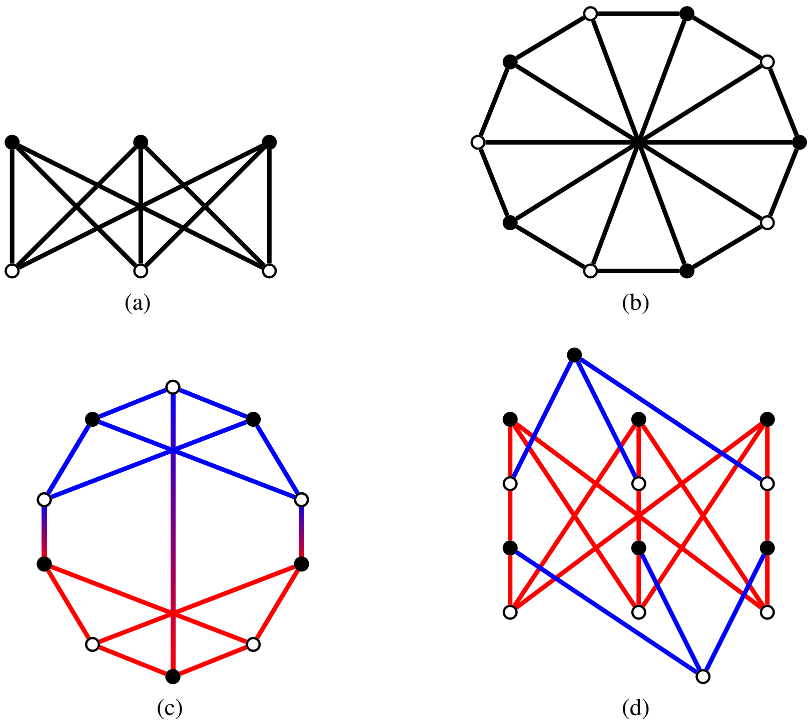


Figure 3.5 – Four cubic bipartite graphs G with $\overline{\chi_\Sigma}(G) = 3$.

Theorem 3.2.7. *There exist infinitely many bipartite graphs G with $2 = \chi_\Sigma(G) < \overline{\chi_\Sigma}(G) = 3$.*

Proof. The proof follows directly by Propositions 3.2.3, 3.2.4, 3.2.5 and 3.2.6. □

We ran computer programs to come up with additional examples of bipartite graphs admitting no equitable proper 2-labellings. We are in particular interested in finding such graphs that cannot be obtained through the application of Operations 1 and 2. We were able to check all cubic bipartite graphs with up to 25 vertices, and all subcubic (non-cubic) bipartite graphs with up to 17 vertices. It turns out that, for this restricted sampling, all bipartite graphs G with $2 = \chi_\Sigma(G) < \overline{\chi_\Sigma}(G) = 3$ are either small, or modifications of these graphs obtained through Operations 1 and 2 described earlier. These graphs are illustrated in Figure 3.5.

All these graphs share particular properties of interest. Notably, they share two important properties of odd multi-cacti, as they are 2-connected and both their partition classes are of odd size. All of these graphs are also related to $K_{3,3}$ somehow. For instance, the graph in Figure 3.5(c) can be obtained by "gluing" two $K_{3,3}$'s one (with red edges) onto the other (with blue edges) (note indeed that by contracting the bottom five vertices to a single vertex (and similarly for the top five vertices), it results in $K_{3,3}$). Also, the graph in Figure 3.5(d) can be obtained by subdividing twice the edges of a perfect matching of $K_{3,3}$ (resulting in the red edges) and joining the resulting vertices to two new degree-3 vertices (via blue edges). In the non-cubic case, we observed that repeatedly applying Operation 1 on the edges of a perfect matching of $K_{3,3}$ yields subcubic bipartite graphs G with $2 = \chi_\Sigma(G) < \overline{\chi_\Sigma}(G) = 3$.

It might be that the four graphs (already mentioned in [113]) depicted in Figure 3.5 are the only cubic bipartite graphs G with $\chi_\Sigma(G) = 2$ and $\overline{\chi_\Sigma}(G) = 3$. Our feeling is that as soon as a cubic bipartite graph is large enough, there should always be several ways to locally alter a proper 2-labelling to make it equitable. As a result, we were unsuccessful in coming up with infinite families of such graphs. For instance, a natural way for generalising the graphs in Figures 3.5(a) and 3.5(b) is as follows: for any $n \equiv 2 \bmod 4$, we denote by $C_n^\times$ the cubic graph obtained from C_n by adding an edge between any two antipodal vertices. Note indeed that the graph in Figure 3.5(a), which is $K_{3,3}$, is also $C_6^\times$, while the graph in Figure 3.5(b) is $C_{10}^\times$. Our experimentations show that, unfortunately, it seems that $\overline{\chi_\Sigma}(C_n^\times) = 2$ whenever $n \geq 14$.

In light of these arguments, let us finish this section by raising the following questions:

Question 3.2.8. *Let G be a bipartite graph with $2 = \chi_\Sigma(G) < \overline{\chi_\Sigma}(G) = 3$.*
- *Can we have $\delta(G) = 1$?*
- *Can we have $\Delta(G) \geq 4$?*
- *Can G have cut vertices?*
- *Can we have $|V(G)| \not\equiv 2 \bmod 4$?*
- *Can G be cubic with $|V(G)|$ being arbitrarily large?*
- *Was G obtained from $K_{3,3}$ by repeatedly applying certain operations?*

3.3 Equitable proper labellings of regular bipartite graphs

In this section, we mainly prove that, for any $k \geq 3$, every k-regular bipartite graph admits equitable proper k-labellings. As a particular case, we get that cubic bipartite graphs form another family of graphs verifying Conjecture 2.1.2.

Our proof makes use of the following result of Kőnig from 1916, which says that regular bipartite graphs are class 1. Recall that a *proper edge-colouring* of a graph is an edge-colouring where no two adjacent edges get assigned the same colour. In other words, the edge-set of any k-regular bipartite graph can be partitioned into k perfect matchings.

Theorem 3.3.1 (Kőnig [87])**.** *All k-regular bipartite graphs admit proper k-edge-colourings.*

We are now ready for our main result.

Theorem 3.3.2. *For all $k \geq 3$, if $G = (A, B, E)$ is a k-regular bipartite graph, then $\overline{\chi_\Sigma}(G) \leq k$.*

Proof. Initially, apply a proper k-edge colouring to G, which exists by Theorem 3.3.1. This initial k-labelling ℓ is equitable, but it is not proper, since every vertex has colour $p = \frac{k(k+1)}{2}$ (each vertex is incident to exactly k edges, each with a unique label from 1 to k). The following four-step algorithm which makes local swaps of labels is applied to the k-labelling ℓ of G, until the k-labelling ℓ becomes proper. Note that the algorithm only swaps labels of edges, and therefore, the k-labelling ℓ remains equitable throughout. In what follows, for a vertex $v \in V(G)$, an edge $uv \in E(G)$, and an integer $i \geq 1$, let $c_\ell^i(v)$ and $\ell^i(uv)$ be the colour of the vertex v and the label of the edge uv respectively, after the $(i-1)^{th}$ iteration (and before the i^{th} iteration) of the current step being considered. Also, the superscript is omitted from the colour notation when the current colour is being mentioned. The algorithm begins as follows:

1. While there exists a subgraph of G isomorphic to $P_3 = (u, x, v)$ such that $u, v \in A$ and $c_\ell(u) = c_\ell(v) = p$, swap $\ell(ux)$ with $\ell(vx)$.

There are three things to note after the first step of the algorithm. The first is that, for each pair of vertices $u, v \in A$ that are dealt with in the i^{th} iteration of Step 1, $c_\ell^{i+1}(u) \neq p$ and $c_\ell^{i+1}(v) \neq p$ for all $i \geq 1$. Indeed, their colours were p before the step was executed but since $\ell(ux) \neq \ell(vx)$ for all these pairs $u, v \in A$ (x is incident to exactly one edge with label j for all $1 \leq j \leq k$), their colours cannot be p after the step is executed. Also, only the labels of edges incident to the vertices u and v are changed and so, the edges whose labels are changed at each execution of Step 1 are all disjoint. The second thing is that, for every vertex $u \in B$, we have $c_\ell(u) = p$. The third thing is that, once Step 1 can no longer be executed, for any two vertices $u, v \in A$ such that $c_\ell(u) = c_\ell(v) = p$, we have that $\mathrm{dist}(u, v) \geq 4$. Now, the algorithm proceeds as follows:

2. While there exists an induced subgraph of G isomorphic to $P_5 = (u, x, z, y, v)$ such that $u, v \in A$ and $c_\ell(u) = c_\ell(v) = p$,
   (a) swap $\ell(ux)$ with $\ell(xz)$ if this results in $c_\ell(z) \neq p$;
   (b) else, swap $\ell(vy)$ with $\ell(yz)$ if this results in $c_\ell(z) \neq p$;
   (c) else, swap $\ell(ux)$ with $\ell(xz)$ and $\ell(vy)$ with $\ell(yz)$.

Claim 3.3.3. *If one of Steps 2(a)-(c) is executed on the i^{th} iteration of Step 2, then $c_\ell^{i+1}(z) \neq p$. Moreover, after any of them is executed, each vertex in B is still incident to exactly one edge with label j for all $1 \leq j \leq k$.*

Proof of the claim. If Step 2(a) or Step 2(b) is executed, then $c_\ell^{i+1}(z) \neq p$ by definition. If Step 2(a) cannot be executed, then $c_\ell^i(z) - \ell^i(xz) + \ell^i(ux) = p$. Observe that $\ell^i(ux) \neq \ell^i(xz)$ and $\ell^i(vy) \neq \ell^i(yz)$ since $x, y \in B$ and each vertex in B is still incident to exactly one edge with label j for all $1 \leq j \leq k$ (trivial induction on the number of times such a process has been performed before this step). Therefore, $c_\ell^i(z) - \ell^i(xz) + \ell^i(ux) - \ell^i(yz) + \ell^i(vy) = p - \ell^i(yz) + \ell^i(vy) \neq p$ and so, $c_\ell^{i+1}(z) \neq p$ if Step 2(c) is executed. $\diamond$

Note that in all cases of Step 2, after its i^{th} execution, either $c_\ell^{i+1}(u) \neq p$ or $c_\ell^{i+1}(v) \neq p$. Moreover, if $c_\ell^{i+1}(u) = p$ ($c_\ell^{i+1}(v) = p$ respectively), then none of the labels of the edges incident to u (v respectively) were changed. Note that Step 2 eventually ends since no new vertices get colour p by Claim 3.3.3 and at least one vertex changes from colour p to another colour after each execution of Step 2. Once Step 2 can no longer be executed, for any two vertices $u, v \in A$ such that $c_\ell(u) = c_\ell(v) = p$, we have that $\mathrm{dist}(u, v) > 4$. The algorithm proceeds as follows:

3. While there exists a subgraph of G isomorphic to $C_4 = (u, x, z, y, u)$ such that $u \in A$ and $c_\ell(u) = p$,
   (a) swap $\ell(ux)$ with $\ell(xz)$ if this results in $c_\ell(z) \neq p$;
   (b) else, swap $\ell(uy)$ with $\ell(yz)$ if this results in $c_\ell(z) \neq p$;
   (c) else, swap $\ell(ux)$ with $\ell(xz)$ and $\ell(uy)$ with $\ell(yz)$.

From a proof analogous to that of Claim 3.3.3, if one of Steps 3(a)-(c) is executed at the i^{th} iteration of Step 3, then $c_\ell^{i+1}(z) \neq p$. Furthermore, in all cases of Step 3, after its i^{th} iteration, we have $c_\ell^{i+1}(u) \neq p$, which is obvious except in the case that Step 3(c) was executed. Note that if $c_\ell^{i+1}(u) = p$ after the i^{th} iteration of Step 3(c), then $c_\ell^i(u) - \ell^i(ux) + \ell^i(xz) - \ell^i(uy) + \ell^i(yz) = p$. In this case, $\ell^i(xz) - \ell^i(ux) = \ell^i(uy) - \ell^i(yz)$, since $c_\ell^i(u) = p$. But then, since Steps 3(a) and 3(b) were not executable, $c_\ell^i(z) - \ell^i(xz) + \ell^i(ux) = c_\ell^i(z) - \ell^i(yz) + \ell^i(uy) = p$, which implies that $\ell^i(ux) - \ell^i(xz) = \ell^i(uy) - \ell^i(yz)$. This is a contradiction since both $\ell^i(xz) - \ell^i(ux) =$

$\ell^i(uy) - \ell^i(yz)$ and $\ell^i(ux) - \ell^i(xz) = \ell^i(uy) - \ell^i(yz)$ hold if and only if $\ell^i(xz) = \ell^i(ux)$, but $\ell^i(xz) \neq \ell^i(ux)$ since $x, y \in B$ and each of the vertices of B is still incident to exactly one edge with label j for all $1 \leq j \leq k$, even after each iteration of Step 3 is executed until Step 3 can no longer be executed. Therefore, Step 3(a) or 3(b) was executable and so, Step 3(c) would not have been executed.

Note that Step 3 eventually ends since no new vertices get colour p and one vertex changes from colour p to another colour after each execution of Step 3. Once Step 3 can no longer be executed, then for any vertex $u \in A$ such that $c_\ell(u) = p$, we have that for any two vertices $x, y \in N(u)$, it holds that $N(x) \cap N(y) = u$. The remainder of the algorithm depends on the value of k with the case where $k = 3$ being different from the case $k \geq 4$. In what follows, we denote by S_x the star with x leaves (being isomorphic to $K_{1,x}$).

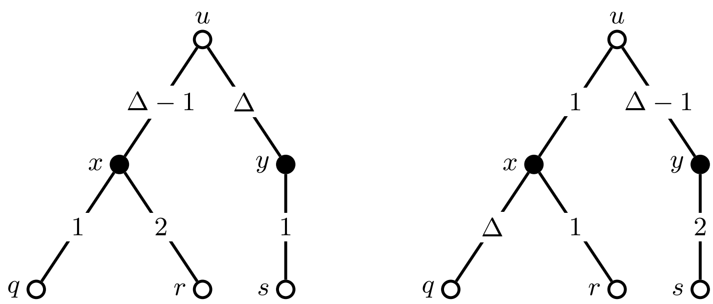
Case $k = 3$: note that $p = 6$ in this case. The algorithm proceeds as follows:

4. While there exists a subgraph of G isomorphic to S_3, with center u and leaves x, y, z such that $u \in A$ and for all $w \in N(u)$, we have that $c_\ell(u) = c_\ell(w) = 6$, then, w.l.o.g., we may assume that $\ell^i(ux) = 1$, $\ell^i(uy) = 2$, and $\ell^i(uz) = 3$, and
   (a) if for some $w \in N(u)$ and some $v \in N(w) \setminus \{u\}$, swapping $\ell(uw)$ with $\ell(wv)$ results in $c_\ell(v) \neq 6$, then swap $\ell(uw)$ with $\ell(wv)$;
   (b) else, for all $q \in N(x) \setminus \{u\}$, for all $r \in N(y) \setminus \{u\}$, and for all $s \in N(z) \setminus \{u\}$, remove the labels of the edges xq, yr, and zs, for a total of six labels removed. Note that two 1s, two 2s, and two 3s have been removed since each vertex in the closed neighbourhood of u is incident to exactly one edge with label j for all $1 \leq j \leq 3$. Then, assuming this is the i^{th} iteration of Step 4, for all $q \in N(x) \setminus \{u\}$, for all $r \in N(y) \setminus \{u\}$, and for all $s \in N(z) \setminus \{u\}$, set $\ell^{i+1}(xq) = 2$, $\ell^{i+1}(yr) = 3$, and $\ell^{i+1}(zs) = 1$.

First, note that only edges incident to vertices at distance at most 2 from u have their labels changed and so each execution of Step 4 deals with disjoint vertices and edges in relation to the other executions of Step 4. If Step 4(a) is executed at the i^{th} iteration, then $c_\ell^{i+1}(u) \neq 6$ and no other vertex changed from colour 6 to another colour or from another colour to 6. If Step 4(b) is executed at the i^{th} iteration, then $c_\ell^{i+1}(u) = 6$ remains unchanged, however, $c_\ell^{i+1}(x) = 5$, $c_\ell^{i+1}(y) = 8$, and $c_\ell^{i+1}(z) = 5$. Moreover, since Step 4(a) was not executed, then for all $q \in N(x) \setminus \{u\}$, after the i^{th} iteration of Step 4 (specifically Step 4(b) was executed), we have $c_\ell^{i+1}(q) \neq c_\ell^{i+1}(x)$ and $c_\ell^{i+1}(q) \neq 6$. Indeed, let $\alpha, \beta = N(x) \setminus \{u\}$. For exactly one $q \in N(x) \setminus \{u\}$, say α, the label of $x\alpha$ was changed from 3 to 2. But since Step 4(a) was not executed, we have that $c_\ell^i(\alpha) - 3 + 1 = 6$ and so, $c_\ell^i(\alpha) = 8$. Therefore, $c_\ell^{i+1}(\alpha) = 7$. Also since Step 4(a) was not executed, we have that $c_\ell^i(\beta) - 2 + 1 = 6$ and so, $c_\ell^i(\beta) = 7$. Since none of the labels incident to β changed, $c_\ell^i(\beta) = c_\ell^{i+1}(\beta) = 7$. Analogously, for all $r \in N(y) \setminus \{u\}$ and for all $s \in N(z) \setminus \{u\}$, we have $c_\ell^{i+1}(r) \neq c_\ell^{i+1}(y)$, $c_\ell^{i+1}(r) \neq 6$, $c_\ell^{i+1}(s) \neq c_\ell^{i+1}(z)$, and $c_\ell^{i+1}(s) \neq 6$. Indeed, it is easy to check that for all $r \in N(y) \setminus \{u\}$ and for all $s \in N(z) \setminus \{u\}$, we have $c_\ell^{i+1}(r) = 7$ and $c_\ell^{i+1}(s) = 4$.

Note that Step 4 eventually ends since either $c_\ell(u) \neq 6$ or all of the neighbours of u have a colour different from 6 after each execution of Step 4, no vertices change to colour 6, and no new vertices come into conflict in terms of colour. Once, Step 4 can no longer be executed, the 3-labelling ℓ is proper and equitable. Indeed, there are no more vertices in A whose colour conflicts with a vertex in B.

Case $k \geq 4$: the algorithm proceeds as follows:



(a) Before Step 4(b) is executed. (b) After Step 4(b) is executed.

Figure 3.6 – The case of the proof of Theorem 3.3.2 before and after Step 4(b) is executed when $k \geq 4$.

4. While there exists a subgraph of G isomorphic to S_k, with center u such that $u \in A$ and for all $w \in N(u)$, we have that $c_\ell(u) = c_\ell(w) = p$ (see Figure 3.6),
   (a) if for some $w \in N(u)$ and some $v \in N(w) \setminus \{u\}$, swapping $\ell(uw)$ with $\ell(wv)$ results in $c_\ell(v) \neq p$, then swap $\ell(uw)$ with $\ell(wv)$;
   (b) else, let $x, y \in N(u)$, $q, r \in N(x) \setminus \{u\}$ and $s \in N(y) \setminus \{u\}$ such that, if this is the i^{th} iteration of Step 4, $\ell^i(ux) = \Delta - 1$, $\ell^i(uy) = \Delta$, $\ell^i(xq) = 1$, $\ell^i(xr) = 2$, $\ell^i(ys) = 1$, and swap the labels of these edges so that $\ell^{i+1}(ux) = 1$, $\ell^{i+1}(uy) = \Delta-1$, $\ell^{i+1}(xq) = \Delta$, $\ell^{i+1}(xr) = 1$, and $\ell^{i+1}(ys) = 2$ (see Figure 3.6). Note that such a labelling ℓ^i exists since each vertex in the closed neighbourhood of u is incident to exactly one edge with label j for all $1 \leq j \leq k$.

First, note that only edges incident to vertices at distance at most 2 from u have their labels changed and so each execution of Step 4 deals with disjoint vertices and edges in relation to the other executions of Step 4. If Step 4(a) is executed at the i^{th} iteration of Step 4, then $c_\ell^{i+1}(u) \neq p$ and no other vertex changed from colour p to another colour or from another colour to p. If Step 4(b) is executed at the i^{th} iteration of Step 4, then $c_\ell^{i+1}(u) = p - \Delta + 1 \neq p$, $c_\ell^{i+1}(x) = p - \Delta + 1 + \Delta - 2 + 1 = p$, $c_\ell^{i+1}(y) = p - \Delta + \Delta - 1 - 1 + 2 = p$, and since Step 4(a) was not executed, $\Delta \neq \Delta - 1$, and $\Delta \neq 2$, we have that $c_\ell^{i+1}(q) \neq p$, $c_\ell^{i+1}(r) \neq p$, and $c_\ell^{i+1}(s) \neq p$.

Note that Step 4 eventually ends since $c_\ell(u) \neq p$ after each execution of Step 4 and no new vertices come into conflict in terms of colour. Once, Step 4 can no longer be executed, the k-labelling ℓ is proper and equitable. Indeed, there are no more vertices in A whose colour conflicts with a vertex in B. □

3.4 Conclusion

In this chapter, we have provided several results on equitable proper labellings, a notion that was previously introduced and studied in [18] and [113]. Answering a question of Senhaji, we proved that there exist infinitely many graphs G with $\chi_\Sigma(G) < \overline{\chi_\Sigma}(G)$. Actually, unless P=NP, deciding if a graph G verifies $\chi_\Sigma(G) = \overline{\chi_\Sigma}(G)$ cannot be done in polynomial time. In the bipartite case, we exhibited operations for building infinitely many bipartite graphs G with $\chi_\Sigma(G) < \overline{\chi_\Sigma}(G)$. We also proved that, for every bipartite graph G with $\chi_\Sigma(G) = 3$, we have

$\overline{\chi_\Sigma}(G) = 3$. Finally, we proved that $\overline{\chi_\Sigma}(G) \leq k$ holds for every k-regular bipartite graph G with $k \geq 3$.

Regarding our results, some aspects remain open.

— In particular, we still wonder whether there is a good characterisation of bipartite graphs G with $2 = \chi_\Sigma(G) < \overline{\chi_\Sigma}(G) = 3$. Recall that all such graphs we have exhibited share very particular properties, which led to raising Question 3.2.8, whose aspects are very intriguing. If such a good characterisation was to not exist, then that would be an interesting contrast with the non-equitable case (regarding the characterisation of odd multi-cacti from [116], presented in Section 2.3.1).

— Regarding Conjecture 2.1.2 (which, recall, claims that for almost all graphs G, $\overline{\chi_\Sigma}(G) \leq 3$), only the case $k = 3$ of our Theorem 3.3.2 gives a satisfying answer. A next step could be to prove Conjecture 2.1.2 for all k-regular bipartite graphs with $k \geq 4$. Recall that Kőnig's Theorem (Theorem 3.3.1) was a nice tool for ensuring equitability in our proof of Theorem 3.3.2.

More generally speaking, there are still fundamental aspects of Conjecture 2.1.2 which we barely understand. In particular, it would be nice to provide any general constant upper bound on $\overline{\chi_\Sigma}$. Providing such a bound even in the bipartite case would already be something. Also, it would be interesting to know whether K_4 is the only connected graph for which $\overline{\chi_\Sigma}$ is more than 3.

CHAPTER 4

Minimising the sum of assigned labels

In this chapter, we study proper labellings of graphs with the extra requirement that the sum of assigned labels must be as small as possible.
We investigate several aspects of this problem, covering algorithmic and combinatorial aspects. In particular, we prove that the problem of designing proper labellings with minimum label sum is NP*-hard in general, but solvable in polynomial time for graphs with bounded treewidth. We also conjecture that for almost every connected graph G there should be a proper labelling with label sum at most $2|E(G)|$, which we verify for several classes of graphs.*
This chapter presents a joint work with J. Bensmail and N. Nisse, published in [29] and presented in [28].

Recall that the problem of finding a proper k-labelling for a given graph G, is equivalent to the problem of finding a locally irregular multigraph M with at most k parallel edges between each pair of vertices, constructed from G by multiplying its edges in a particular fashion. In their work introducing the notion of the irregularity strength of a graph (where M is required to be irregular rather than locally irregular), the authors of [50] regard edge multiplications as an expensive operation and, as such, they want to limit it as much as possible.

Minimising, however, the maximum label that is used to create a proper labelling of a graph G, does not always guarantee that we have actually minimised the cost that corresponds to the edge multiplications described above. For example, a 2-labelling ℓ that assigns label 2 to three edges of G and label 1 to the rest is more expensive (in terms of how many edges the corresponding multigraphs have) than a 3-labelling ℓ' that assigns label 3 to only one edge of G and label 1 to the rest. The 1-2-3 Conjecture, if true, would imply that every nice graph G admits a proper labelling where the sum of assigned labels is at most $3|E(G)|$; but it might be that, using labels with value larger than 3, we can design better (with respect to the concerns above) proper labellings of G.

In this chapter we study the problem of finding proper labellings that also minimise the sum of labels being used. Recall that, for a given nice graph G, the parameter $\mathrm{mE}(G)$ is the minimum sum of labels assigned to the edges of G by any proper labelling of G.

Determining $\mathrm{mE}(G)$ for a given graph G is also related to finding a proper labelling of G where the resulting vertex colours satisfy some properties. More precisely, by a straight equivalence between edge labels and vertex colours, see upcoming Observation 4.1.3, it can be established that determining $\mathrm{mE}(G)$ is equivalent to finding a proper labelling of G that minimises the sum of resulting vertex colours. Thus, at least at first glance, one could think that determining $\mathrm{mE}(G)$ is somewhat related to finding proper labellings whose induced colourings verify some specific properties in regards to a proper vertex-colouring of G. For example, there could be links with the investigations [16] and [31], where the authors study proper labellings with a minimum number of induced colours or that minimise the maximum induced colour respectively. In Section 4.3.1, we actually show that this is not the case, in the sense that proper labellings that are good for our concerns might be arbitrarily bad for those in [16] and [31], and *vice versa*.

In Section 4.1, we provide some observations that will be used throughout this chapter. As a warm up, we also provide the exact value of $\mathrm{mE}(G)$ for simple classes of graphs G, namely complete bipartite graphs, complete graphs and cycles.

In Section 4.2, we deal with the algorithmic aspects of the problem. Recall that for $k \in \mathbb{N}$ and a graph G, $\mathrm{mE}_k(G)$ is the minimum sum of assigned labels over all proper k-labellings of G. We show that for $k \in \mathbb{N}$, determining $\mathrm{mE}_k(G)$ is NP-complete when G is a planar bipartite graph. Then we provide an algorithm that, given two integers s and k, decides in polynomial time if $\mathrm{mE}_k(G) \leq s$ when G belongs to the family of graphs that have bounded treewidth.

In Section 4.3, we answer two different questions that deal with the particular nature of our problem. First, in Section 4.3.1, we show that in general a proper k-labelling that minimises the maximum induced colour does not minimise the sum of the labels used, and *vice versa*. Then, in Section 4.3.2, we provide an infinite family of graphs G for which $\mathrm{mE}_k(G)$, for every $k \geq 2$, can be arbitrarily larger than $\mathrm{mE}_{k+1}(G)$. As mentioned earlier, this property justifies the study of our problem, as it shows that just finding a proper k-labelling of G minimising the sum of labels for $k = \chi_\Sigma(G)$ is not enough.

Finally, in Section 4.4, we study more general aspects of the problem. In particular, we propose Conjecture 2.1.4 stating that for every nice connected graph G, we should have $\mathrm{mE}(G) \leq$

Graphs	**Upper bound**
$\chi(G) = k \geq 3$	$\mathrm{mE}_{k+1}(G) \leq \lvert E(G)\rvert + k\lvert V(G)\rvert$
$\chi(G) = k \geq 3$ and k odd	$\mathrm{mE}_k(G) \leq \lvert E(G)\rvert + (k-1)\lvert V(G)\rvert$
Regular and $\chi_\Sigma(G) = 2$	$\mathrm{mE}_2(G) \leq \frac{3}{2}\lvert E(G)\rvert$
Bipartite	$\mathrm{mE}_3(G) \leq 2\lvert E(G)\rvert$
Bipartite and one bipartition class of even size	$\mathrm{mE}_2(G) \leq \lvert E(G)\rvert + \lvert V(G)\rvert - 1$
Hamiltonian bipartite and one bipartition class of even size	$\mathrm{mE}_2(G) \leq \frac{3}{2}\lvert E(G)\rvert$
Tree	$\mathrm{mE}_2(G) \leq \frac{3}{2}\lvert E(G)\rvert$

Table 4.1 – Summary of the upper bounds presented in Section 4.4. We also provide a construction producing infinitely many connected bipartite graphs G such that $\mathrm{mE}_2(G) = \frac{3}{2}|E(G)|$.

$2|E(G)|$. We then proceed by providing upper bounds for some families of graphs, namely bipartite graphs and trees, as well as graphs with large chromatic number, that further strengthen our belief that Conjecture 2.1.4 should hold true. In Table 4.1 we summarise the results presented in this section.

4.1 First observations and classes of graphs

In this warm-up section, we give some first insight into the problem of determining the parameters $\mathrm{mE}(G)$ and $\mathrm{mE}_k(G)$ for a given graph G. This is done through first observations on the problem, and by then focusing on simple classes of graphs.

4.1.1 First observations and remarks

Let ℓ be a proper k-labelling of a graph G. Recall that $\sigma(\ell)$ is the sum of the labels assigned by ℓ. Now since each edge of G is assigned a label between 1 and k, the next trivial bounds follow directly:

Observation 4.1.1. *Let $G = (V, E)$ be a graph and ℓ be a k-labelling of G. Then*

$$|E| \leq \sigma(\ell) \leq k|E|.$$

Consequently, for any $k \geq \chi_\Sigma(G)$,

$$|E| \leq \mathrm{mE}_k(G) \leq k|E|.$$

Observation 4.1.1 establishes that, for any nice graph G, in general $\mathrm{mE}(G)$ should be expressed as a function of $|E(G)|$. As far as direct upper bounds are concerned, recall that for every nice graph, $\chi_\Sigma(G) \leq 5$ [79]. This implies that:

Theorem 4.1.2 ([79])**.** *For every nice graph $G = (V, E)$, $\mathrm{mE}(G) \leq \mathrm{mE}_5(G) \leq 5|E|$.*

Of course, the upper bound in Theorem 4.1.2 is immediately improved for every nice graph G for which the upper bound on $\chi_\Sigma(G)$ can be improved. In particular, recall that $\chi_\Sigma(G) \leq 3$ whenever $\chi(G) \leq 3$ (see [81]), which implies that $\mathrm{mE}(G) \leq \mathrm{mE}_3(G) \leq 3|E(G)|$ holds for such graphs. Recently, Przybyło proved in [105] that $\chi_\Sigma(G) \leq 4$ whenever G is regular, which implies that $\mathrm{mE}(G) \leq \mathrm{mE}_4(G) \leq 4|E(G)|$ holds for regular graphs. More results of this sort can be found in [112].

We close this section with the following observation, which provides an obvious way for relating edge labels and vertex colours by a labelling of a graph G. In general, this observation is a convenient tool for establishing lower bounds on $\mathrm{mE}(G)$.

Observation 4.1.3. *Let $G = (V, E)$ be a graph and ℓ be a labelling of G. Then*

$$\sum_{e \in E} 2\ell(e) = \sum_{v \in V} c_\ell(v).$$

In particular, by any labelling ℓ, the sum $\sum_{v \in V} c_\ell(v)$ must be an even number.

4.1.2 Simple classes of graphs

In this section, we determine the value of $\mathrm{mE}(G)$ when G is any nice complete bipartite graph, complete graph, or cycle. Let us recall that, for any nice complete bipartite graph $K_{n,m}$, we have $\chi_\Sigma(K_{n,m}) = 1$ if $n \neq m > 1$, and $\chi_\Sigma(K_{n,m}) = 2$ otherwise. For every nice complete graph K_n, we have $\chi_\Sigma(K_n) = 3$. For every nice cycle C_n, we have $\chi_\Sigma(C_n) = 3$ whenever $n \geq 3$ is odd or $n \equiv 2 \bmod 4$, while we have $\chi_\Sigma(C_n) = 2$ otherwise, *i.e.*, when $n \equiv 0 \bmod 4$. Simple proofs for these statements can be found *e.g.* in [50, 46].

Note that in all the results obtained in this section, constructing a proper labelling ℓ of a graph G achieving $\sigma(\ell) = \mathrm{mE}(G)$ does not require the use of a label larger than $\chi_\Sigma(G)$. That is, we here always have $\mathrm{mE}_k(G) = \mathrm{mE}(G)$ for $k = \chi_\Sigma(G)$. It is important to point out however that this behaviour is not true in general (see Section 4.3.2).

Theorem 4.1.4. *Let $G = (A, B, E) = K_{n,m}$ be a complete bipartite graph with $n + m > 2$. Then:*

— *if $n \neq m$, then $\mathrm{mE}(G) = \mathrm{mE}_1(G) = nm = |E|$;*
— *otherwise,* i.e.*, $n = m$, we have $\mathrm{mE}(G) = \mathrm{mE}_2(G) = n(m+1) = |E| + \sqrt{|E|}$.*

Proof. If $n \neq m$, then G is locally irregular, in which case we get a proper 1-labelling when assigning label 1 to all edges. This is best possible due to Observation 4.1.1. If $n = m$, then G is not locally irregular, which implies that a proper labelling of G must assign a label different from 1 to some edges. Moreover, if a labelling assigns a label different than 1 to less than n edges, then there would necessarily be, in both A and B, vertices incident only to edges labelled 1, thus with colour n. In that case, ℓ would not be proper as some adjacent vertices would have the same colour.

This means that if $n = m$, a proper labelling ℓ of G must assign a label different from 1 to at least n edges. This implies that $\mathrm{mE}_k(G) \geq |E| + n$. We claim there is a proper 2-labelling ℓ achieving this lower bound, hence best possible. To obtain ℓ, let a be any vertex of A. Assign label 2 to all the n edges incident to a, and assign label 1 to all other edges. This labelling is proper. Indeed $c_\ell(a) = 2n$, $c_\ell(a') = n$ for every $a' \in A \setminus \{a\}$, and $c_\ell(b) = n+1$ for every $b \in B$. Furthermore, $\sigma(\ell) = |E| + n = |E| + \sqrt{|E|}$. □

Theorem 4.1.5. *Let $K_n = (V, E)$ be a complete graph with $n \geq 3$. Then:*

— *if $n = 3$, then $\mathrm{mE}(K_3) = \mathrm{mE}_3(K_3) = 6 = 2|E|$;*

— *if* $n \equiv 0 \bmod 4$ *or* $n \equiv 1 \bmod 4$*, then* $\mathrm{mE}(K_n) = \mathrm{mE}_3(K_n) = \frac{1}{2}\left(n^2 + \frac{(n-2)(n-1)}{2} - 1\right) = \frac{3}{2}|E|$*;*

— *if* $n \equiv 2 \bmod 4$ *or* $n \equiv 3 \bmod 4$*, then* $\mathrm{mE}(K_n) = \mathrm{mE}_3(K_n) = \frac{1}{2}\left(n^2 + \frac{(n-2)(n-1)}{2}\right) = \left\lceil \frac{3}{2}|E| \right\rceil$.

Proof. Throughout this proof, for any $n \geq 3$, let $V = \{v_1, \dots, v_n\}$.

Regarding the first item, Observation 2.1.5 implies that a proper 3-labelling of K_3 must assign three distinct labels to the edges, and thus having $\{\ell(v_1v_2), \ell(v_1v_3), \ell(v_2v_3)\} = \{1, 2, 3\}$ is optimal, in which case $\sigma(\ell) = 6$.

Let us now focus on the second and third items. Following Observation 4.1.3, finding a proper 3-labelling of K_n achieving $\mathrm{mE}_3(K_n)$ is equivalent to finding a proper 3-labelling minimising the sum of vertex colours. Since, in K_n, all vertices have degree $n-1$, and all vertex colours must be different by a proper 3-labelling, any proper 3-labelling producing distinct vertex colours in $S_1 = \{n-1, n, n+1, \dots, 2n-2\}$ would be optimal. Note, however, that when n is congruent to 2 or 3 modulo 4, such a proper 3-labelling cannot exist as, in such cases, the sum $n-1+n+(n+1)+\dots+(2n-2)$ of the values in S_1 is odd, which cannot be achieved by a labelling (recall the last statement of Observation 4.1.3). In these cases, however, any proper labelling producing distinct vertex colours in $S_2 = \{n-1, n, n+1, \dots, 2n-3, 2n-1\}$ would be optimal.

Now consider the following 3-labelling ℓ of K_n ($n \geq 4$), already introduced in [31], to establish what the value of $\mathrm{mS}_3(K_n)$ is (where $\mathrm{mS}_3(G)$ denotes the smallest maximum color over the vertices by a proper 3-labelling of G). We label the edges of K_n through three steps. Firstly, we assign label 1 to every edge. Secondly, we change the labels of the edges in $\{v_iv_j : 1 \leq i, j \leq n, i + j \geq n + 2\}$ to 2. Then v_1 is incident to no edge labelled 2, vertex v_2 is incident to one edge labelled 2, vertex v_i for $3 \leq i \leq \lfloor (n-1)/2 \rfloor + 1$ is incident to $i-1$ edges labelled 2, and v_i for $\lfloor (n-1)/2 \rfloor + 2 \leq i \leq n$ is incident to $i-2$ edges labelled 2. Let $j = \lfloor (n-1)/2 \rfloor + 1$. Note that for every $i \in \{2, 3, \dots, j, j+2, \dots, n\}$, v_i is adjacent to one more edge labelled 2 than v_{i-1}; and that v_j and v_{j+1} are both adjacent to $j-1$ edges labelled 2 (and $n-j$ edges labelled 1). So $c_l(v_1) < c_l(v_2) < \dots < c_l(v_j) = c_l(v_{j+1}) < c_l(v_{j+2}) < \dots < c_l(v_n)$ and $c_l(v_{i+1}) \leq c_l(v_i) + 1$ for $1 \leq i \leq n$, *i.e.*, all vertices have different colours except v_j and v_{j+1}. Finally, to avoid the conflict between v_j and v_{j+1}, let us increase the label of $v_{j+1}v_{j+2}$ from 2 to 3. This change induces a new conflict between v_{j+2} and v_{j+3}. Then we need to increase the label of $v_{j+3}v_{j+4}$ from 2 to 3 to get rid of this conflict, which creates a new conflict, and so on. Formally, we change the labels of the edges in $\{v_{j+1}v_{j+2}, v_{j+3}v_{j+4}, \dots, v_{n-1}v_n\}$ to 3 if $n-j$ is even, *i.e.*, if $n \equiv 0 \mod 4$ or $n \equiv 1 \mod 4$. Otherwise, if $n-j$ is odd and $n \equiv 2 \mod 4$ or $n \equiv 3 \mod 4$, then we change the labels of the edges in $\{v_{j+1}v_{j+2}, v_{j+3}v_{j+4}, \dots, v_{n-4}v_{n-3}, v_{n-2}v_n, v_{n-1}v_n\}$ to 3.

It can be checked that the resulting 3-labelling ℓ is proper, and achieves vertex colours in S_1 when n is congruent to 0 or 1 modulo 4, or vertex colours in S_2 when n is congruent to 2 or 3 modulo 4. As discussed above, this is best possible. Furthermore, it can easily be checked that the elements in S_1 sum up to the value claimed in the second item, and similarly for the elements in S_2 and the value claimed in the third item. This concludes the proof. □

Theorem 4.1.6. *Let $n \geq 3$, and $C_n = (V, E)$ be the cycle of length n. Then:*

— *if* $n \equiv 0 \bmod 4$*, then* $\mathrm{mE}(C_n) = \mathrm{mE}_2(C_n) = \frac{3}{2}|E|$*;*

— *if* $n \equiv 1 \bmod 4$ *or* $n \equiv 3 \bmod 4$*, then* $\mathrm{mE}(C_n) = \mathrm{mE}_3(C_n) = \left\lceil \frac{3}{2}|E| \right\rceil + 1$*;*

— *if* $n \equiv 2 \bmod 4$*, then* $\mathrm{mE}(C_n) = \mathrm{mE}_3(C_n) = \frac{3}{2}|E| + 3$.

Proof. Let us order the edges of C_n following a "clockwise direction" and define $E = \{e_1, \dots, e_n\}$ and $V = \{v_1, \dots, v_n\}$ such that for $i < n$, $e_i = v_i v_{i+1}$ and $e_n = v_n v_1$. Thus, for $i > 0$, $N(v_i) = \{v_{i-1}, v_{i+1}\}$ and $N(v_1) = \{v_2, v_n\}$. Recall that $\chi_\Sigma(C_n) = 2$ for the first item and that $\chi_\Sigma(C_n) = 3$ for the second and third items.

Claim 4.1.7. *Let $l \le k$ and ℓ be a k-labelling of C_n that assigns label l to at least one edge. If ℓ is proper, then it assigns label l to at most $\lfloor \frac{1}{2}|E| \rfloor$ edges if n is odd, while it assigns label l to at most $\frac{1}{2}|E| - 1$ edges if $n \equiv 2 \bmod 4$, and it assigns label l to at most $\frac{1}{2}|E|$ edges if $n \equiv 0 \bmod 4$.*

Proof of the claim. Let $\mathcal{E} = \{e \in E : \ell(e) = l\}$ and $G = (V', E')$ be the graph that has $V' = \{v_i : e_i \in E\}$ and, for $i \neq j$, $v_i v_j \in E'$ if the corresponding edges e_i, e_j are at distance exactly 2 in C_n. Obviously $|E| = |V'|$. It follows from Observation 2.1.5 that if ℓ is a proper labelling of C_n that maximises $|\mathcal{E}|$, then $|\mathcal{E}| = |S|$, where S is an independent set of G. For n odd, G is a copy of the graph C_n. Since G is a cycle, $|\mathcal{E}| = |S| = \lfloor \frac{1}{2}|V'| \rfloor = \lfloor \frac{1}{2}|E| \rfloor$. For $n \equiv 2 \bmod 4$, let m be such that $n = 4m + 2$. It is clear that G contains two connected components, each one being a copy of the cycle $C_{\frac{n}{2}}$. Thus, $|\mathcal{E}| = |S| = 2\lfloor \frac{n}{4} \rfloor = 2m = \frac{1}{2}|E| - 1$. Similarly, if $n = 4m$, it is clear that G contains two connected components, each one being a copy of the cycle $C_{\frac{n}{2}}$. Thus, $|\mathcal{E}| = |S| = 2m = \frac{1}{2}|E|$. $\diamond$

We are now ready to deal with the four values of n (modulo 4) separately:

— For the first item let ℓ be the following 2-labelling: $\ell(e_1) = 1, \ell(e_2) = 1, \ell(e_3) = 2, \ell(e_4) = 2, \ell(e_5) = 1, \dots, \ell(e_n) = 2$. Let us assume that this ℓ is not proper. Then there would exist at least two adjacent vertices v_i, v_{i+1} such that $c_\ell(v_i) = c_\ell(v_{i+1})$. It follows that $\ell(v_{i-1}) = \ell(v_{i+2})$ (if $i = 1$ then $v_{i-1} = v_n$ and if $i = n$ then $v_{i+1} = v_1$) which is a contradiction. Furthermore, since $n \equiv 0 \bmod 4$, label 2 is used on exactly half the edges of C_n and thus $\sigma(\ell) = |E| + \frac{1}{2}|E| = \frac{3}{2}|E|$. Moreover, this value is optimal. Indeed, assume it is not. Then, there would exist a proper labelling ℓ' such that more than $\frac{1}{2}|E|$ edges are labelled 1 by ℓ', a contradiction by Claim 4.1.7.

— Let C_n be a cycle with $n \equiv 1 \bmod 4$. We will show that $\mathrm{mE}(C_n) = \lceil \frac{3}{2}|E| \rceil + 1$. Let ℓ be a proper labelling of C_n that assigns label 3 to only one edge. It follows from Claim 4.1.7 that at most $\lfloor \frac{n}{2} \rfloor$ edges of C_n are labelled 1. Actually, there are exactly $\lfloor \frac{n}{2} \rfloor$ edges labelled 1: if this was not the case, then, since only one edge of C_n is labelled 3, there would be more than $\lfloor \frac{n}{2} \rfloor$ edges labelled 2, contradicting Claim 4.1.7. The same holds true for the edges labelled 2. Since $n \equiv 1 \bmod 4$ implies that there exists an m verifying $n = 4m + 1$, then, using this, one can easily show that $\sigma(\ell) = \lfloor \frac{n}{2} \rfloor + 2\lfloor \frac{n}{2} \rfloor + 3 = \dots = \lceil \frac{3}{2}|E| \rceil + 1$. Furthermore, let ℓ' be a proper labelling of C_n that assigns label 3 to more than one edge. It is clear that if ℓ' is proper, then $\sigma(\ell) < \sigma(\ell')$. Thus, $\sigma(\ell) = \mathrm{mE}(C_n)$. The following is a proper labelling ℓ that achieves this optimal value: $\ell(e_1) = 1, \ell(e_2) = 1, \ell(e_3) = 2, \ell(e_4) = 2, \ell(e_5) = 1, \dots, \ell(e_{n-1}) = 2, \ell(e_n) = 3$.

— Let C_n be a cycle with $n \equiv 3 \bmod 4$. Similarly to before, $\mathrm{mE}(C_n) = \lceil \frac{3}{2}|E| \rceil + 1$. The following is a proper labelling ℓ that achieves this optimal value: $\ell(e_1) = 1, \ell(e_2) = 1, \ell(e_3) = 2, \ell(e_4) = 2, \ell(e_5) = 1, \dots, \ell(e_{n-3}) = 2, \ell(e_{n-2}) = 1, \ell(e_{n-1}) = 3, \ell(e_n) = 2$.

— Let C_n be a cycle with $n \equiv 2 \bmod 4$ ($n \ge 6$). We will show that $\mathrm{mE}(C_n) = \frac{3}{2}|E| + 3$. Indeed, let ℓ be a proper labelling of C_n that assigns label 3 to only one edge. Since ℓ is proper, it is obliged to assign label 1 to at most $\frac{1}{2}|E| - 1$ edges of C_n and label 2 to the

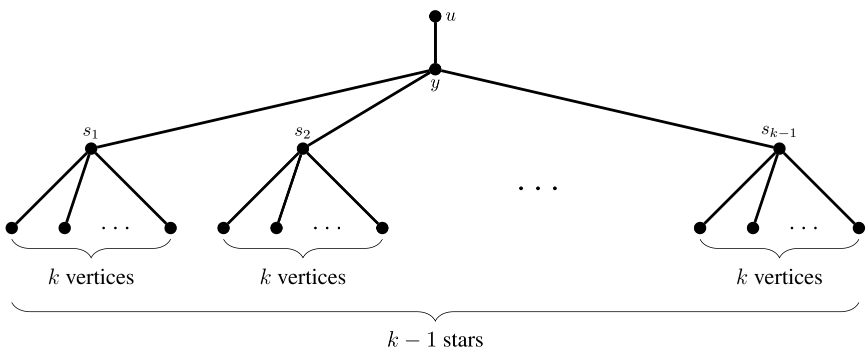


Figure 4.1 – The k-gadget, used in the proof of Theorem 4.2.2

rest. This however would lead to ℓ assigning label 2 on at least $\frac{1}{2}|E|$ edges, which is a contradiction to Claim 4.1.7. Thus, ℓ must assign label 3 on at least two edges. Similarly to before, a labelling that assigns label 3 to exactly two edges, label 1 to at most $\frac{1}{2}|E| - 1$ edges, and label 2 to the rest of the edges, would achieve the optimal value. The following ℓ is one such proper labelling: $\ell(e_1) = 1, \ell(e_2) = 1, \ell(e_3) = 2, \ell(e_4) = 2, \ell(e_5) = 1, \dots, \ell(e_{n-3}) = 2, \ell(e_{n-2}) = 2, \ell(e_{n-1}) = 3, \ell(e_n) = 3$.

□

4.2 Complexity aspects

In this section, we establish both a negative and a positive result on the complexity of computing the parameter $\mathrm{mE}_k(G)$ for some input integer k and nice graph G. More precisely:

— We first prove that determining $\mathrm{mE}_2(G)$ is NP-complete, even when G is restricted to a planar bipartite graph. Recall (from our discussion in Section 2.3.1) that this is contrasting with the complexity of determining whether $\chi_\Sigma(G) \leq 2$ holds for a given bipartite graph G, which is a problem that can be solved in polynomial time due to a result of Thomassen, Wu and Zhang [116].

— We then prove that determining $\mathrm{mE}_k(G)$ can be done in polynomial time whenever k is fixed and G is a graph with bounded treewidth.

4.2.1 A negative result for bipartite graphs

In this section, we prove, in Theorem 4.2.2 below, that the problem of determining $\mathrm{mE}_k(G)$ is NP-complete in planar bipartite graphs G.

Let us first introduce the *k-gadget*, for $k \geq 2$, which will be useful for proving Theorem 4.2.2. To build this gadget, illustrated in Figure 4.1, let us start with $k-1$ *stars*, each having a *center* s_i with $d(s_i) = k+1$ for every $i \in \{1, \dots, k-1\}$. For each star, pick an arbitrary edge $s_i y_i$ and identify all the y_i's into a single vertex y, which is called the *representative* of the gadget. Finally add another vertex u, called the *root* of the gadget, and join it to y via an edge. It is clear that $d(u) = 1$ and $d(y) = k$. Each k-gadget is a tree with $\mathcal{O}(k^2)$ edges. Let v be a vertex of a graph G, and H be a k-gadget. The operation of adding H to G and identifying the root u of H with v is called *attaching* H to v.

Claim 4.2.1. *Let $G = (V, C, E)$ be a bipartite graph and ℓ be any proper 2-labelling of G such that $\sigma(\ell) \leq |E| + c$, for $c = |C|$. Let H be any p-gadget attached to any one vertex of G, where $p - 1 > c$, forming the graph G'. Let y be the representative of H. Let ℓ' be a labelling of G' such that for all edges $e \in E(G)$, we have $\ell'(e) = \ell(e)$. If at least one edge e of H incident to y is labelled 2 by ℓ', then there are at least two edges of H that are labelled 2 by ℓ'.*

Proof of the claim. Let us suppose that at least one of the edges of H incident to y is labelled 2. Let $z \in V \cup C$ be the vertex of G' to which H has been attached.

Let us first assume that the edge zy is labelled 2. If y is incident to only a single edge labelled 2 (*i.e.*, zy), then its colour is $c_\ell(y) = p + 1$. Since all its $p - 1$ neighbours (different from z) have degree $p + 1$, each of them must be incident to at least one edge labelled 2 as otherwise it would have the same colour as y. This leads to at least $p > c$ edges labelled 2, which is a contradiction (observe that since $\sigma(\ell) \leq |E| + c$, there are at most c edges labelled 2). Otherwise, if y has exactly one other incident edge (different from yz) labelled 2, say the edge yw, then we are done.

Then, let us assume that $\ell(yz) = 1$. Moreover, let us assume that some edge incident to y, say yw_1 different from yz, is labelled 2. Then $c_\ell(y) \geq p + 1$ with $c_\ell(y) = p + 1$ if yw_1 is the unique edge incident to y labelled 2. In this case, each one of the $p - 2$ neighbours of y (different from z and w_1) must be incident to at least one edge labelled 2, leading to at least $p - 1 > c$ edges labelled 2, which is a contradiction. Thus $c_\ell(y) > p + 1$, which means there is at least one more edge yw_2 incident to y labelled 2. $\diamond$

We are now ready for proving our result.

Theorem 4.2.2. *Let G be a nice planar bipartite graph, $k \geq 2$ and $q \in \mathbb{N}$. The problem of deciding if $\mathrm{mE}_k(G) \leq q$ is* NP*-complete.*

Proof. The problem is clearly in NP. We focus on showing it is also NP-hard. The proof is done by reduction from PLANAR MONOTONE 1-IN-3 SAT, which was shown to be NP-complete in [100]. In this problem, a 3CNF formula F is given as input, which has clauses with exactly three distinct variables all of which appear only positively. We say that a bipartite graph $G' = (V, C, E)$ *corresponds* to F if it is constructed from F in the following way: for each variable x_i of F add a *variable vertex* v_i in V and for each clause C_j of F add a *clause vertex* c_j in C. Then the edge $v_i c_j$ is added if variable x_i appears in clause C_j. Furthermore, F is valid as input to the PLANAR MONOTONE 1-IN-3 SAT problem if the graph G' that corresponds to F is planar. The question is whether there exists a 1-*in*-3 *truth assignment* of F. Recall that according to such an assignment, each clause has exactly one variable with the value *true*.

Observe now that we may assume that each variable appears in at least two clauses. If there exists a variable, say x_i, that belongs to a single clause $C = (x_i \vee x_j \vee x_k)$, let us add another clause C' identical to C. Clearly, the obtained formula F' is 1-in-3 satisfiable if and only if F is. Moreover, the graph corresponding to F' is planar. Indeed, consider a planar embedding of the graph $G' \setminus \{v_i\}$. Clearly, v_j and v_k are in a same face (since their common neighbour, corresponding to C, has degree 2 in $G' \setminus \{v_i\}$). The graph obtained by adding a vertex v_i (adjacent to the vertex corresponding to C) and a vertex corresponding to C' (adjacent to v_i, v_j and v_k) in this face is planar.

Let us prove the statement for $k = 2$. Let F be a 3CNF formula with c clauses that is given as input to the PLANAR MONOTONE 1-IN-3 SAT problem. Our goal is to construct a planar bipartite graph G such that F is 1-in-3 satisfied if and only if $\mathrm{mE}_2(G) \leq |E(G)| + c$.

Start from $G' = (V, C, E)$ being the planar bipartite graph that corresponds to F, with V being the set of the variable vertices v_i and C being the set of the clause vertices c_j. Note that in F, each clause has exactly three variables but there is no bound on how many times a variable appears in F. Thus for each $v_i \in V$, we have $d(v_i) \geq 2$ and for each $c_j \in C$, we have $d(c_j) = 3$. It follows that $|C| = c$ and $|V| \leq 3c$.

Proceed by modifying G' by adding the gadgets described earlier as follows. For each variable vertex v_i, let d_i be the initial degree of v_i in G'. Let $d_{v,i} = (d_i-1)(c+1)+d_i$ and $d_c = 3(c+1)+3$. For each variable vertex v_i, for all $1 \leq j < d_i$, attach $c+1$ copies of the $(d_{v,i}+j)$-gadget. On each clause vertex c_j, attach $c+1$ copies of the d_c-gadget, $c+1$ copies of the (d_c+2)-gadget and $c+1$ copies of the (d_c+3)-gadget. Name the resulting graph G and observe that the degree of each v_i in G becomes equal to $d_{v,i}$ and the degree of each c_j in G becomes equal to d_c. Clearly, the construction of G is achieved in polynomial time. Finally observe that since G' is planar and the attached gadgets are actually trees, G is also planar.

Let ℓ be a proper 2-labelling of G such that $\sigma(\ell) \leq |E(G)| + c$, *i.e.*, there are at most c edges of G labelled 2 by ℓ. Observe that G contains p-gadgets for $p \in \{d_{v,i}+1, d_{v,i}+2, \dots d_{v,i}+d_i-1, d_c, d_c+2, d_c+3\}$ and $d_{v,i}-1, d_c-1 > c$. Thus Claim 4.2.1 holds for each gadget attached to G.

Claim 4.2.3. *For any proper 2-labelling ℓ of G such that $\sigma(\ell) \leq |E(G)| + c$, we have that:*

— $c_\ell(v_i) \notin \{d_{v,i}+1, d_{v,i}+2, \dots, d_{v,i}+d_i-1\}$ *for each variable vertex* $v_i \in V$*;*

— $c_\ell(c_j) \notin \{d_c, d_c+2, d_c+3\}$ *for each clause vertex* $c_j \in C$.

Proof of the claim. Indeed, each variable vertex v_i is adjacent to $c+1$ copies of the $(d_{v,i}+1)$-gadget and at most c edges are labelled 2 by ℓ. Thus, at least one of the $(d_{v,i}+1)$-gadgets, let us call it H, that is attached to v_i, has all of its edges labelled 1. Moreover, v_i is adjacent to the representative y of H which has degree $d(y) = d_{v,i}+1$. Since all the edges of H are labelled 1, the colour $c_\ell(y)$ of y is $d_{v,i}+1$ and thus this colour is forbidden for v_i, *i.e.*, $c_\ell(v_i) \neq d_{v,i}+1$.

By repeating the same arguments for the $(d_{v,i}+2)$-gadgets attached to v_i, we deduce that $c_\ell(v_i) \neq d_{v,i}+2$. Similarly, by considering the d_c-gadgets (resp., the (d_c+2)- and (d_c+3)-gadgets) attached to any clause vertex c_j, we get that $c_\ell(c_j) \notin \{d_c, d_c+2, d_c+3\}$. ◇

Claim 4.2.4. *Let ℓ be any proper 2-labelling of G such that $\sigma(\ell) \leq |E(G)| + c$. Then all edges of the attached gadgets must be labelled* 1.

Proof of the claim. Observe that for each clause vertex $z \in C$, at least one of its incident edges must be labelled 2. If this were not the case, then $c_\ell(z) = d_c$, and this is not allowed due to Claim 4.2.3.

Let H be a gadget attached to z, and y be the representative of H. Suppose $\ell(yz) = 2$. It follows from Claim 4.2.1 that there are at least two edges of H labelled 2. Recall that the number of edges of G that can be labelled 2 is at most c. Thus, the number of edges of G, that do not belong to H and can be labelled 2, is at most $c-2$. Furthermore, there are $c-1$ clause vertices in G that are different from z. It follows that there exists a clause vertex that has all of its incident edges labelled 1, a contradiction. Thus, each $z \in C$ must be incident to an edge wz with $\ell(wz) = 2$ and w cannot belong to a gadget attached to z. It follows that there must be $|C| = c$ edges of G' labelled 2 and since $\sigma(\ell) \leq |E(G)| + c$, all the edges of the attached gadgets are labelled 1. ◇

It follows from Claim 4.2.4, that the only possible colours induced on the vertices of G' by a proper 2-labelling ℓ of G are:

— $c_\ell(v_i) \in \{d_{v,i}, d_{v,i}+1, d_{v,i}+2, \dots, d_{v,i}+d_i-1, d_{v,i}+d_i\}$ for each variable vertex $v_i \in V$,
— $c_\ell(c_j) \in \{d_c, d_c+1, d_c+2, d_c+3\}$ for each clause vertex $c_j \in C$.

The following hold due to Claim 4.2.3:

— For every variable vertex v_i, we have $c_\ell(v_i) \in \{d_{v,i}, d_{v,i}+d_i\}$. Observe that $c_\ell(v_i) = d_{v,i}$ if all edges of G' adjacent to v_i are labelled 1, and $c_\ell(v_i) = d_{v,i} + d_i$ if all edges of G' adjacent to v_i are labelled 2.
— For every clause vertex c_j, we have $c_\ell(c_j) = d_c + 1$, which corresponds to two edges of G' adjacent to c_j labelled 1 and only one edge labelled 2.

We are now ready to show the equivalence between finding a 1-in-3 truth assignment ϕ of F and finding a proper 2-labelling ℓ of G such that $\sigma(\ell) = \mathrm{mE}_2(G) \leq |E(G)| + c$. An edge v_ic_j of G' labelled 2 (respectively 1) by ℓ corresponds to variable x_i bringing truth value *true* (respectively *false*) to clause C_j by ϕ. Also, we know that in G', every variable vertex v_i is incident to $n \geq 1$ edges, all having the same label (either 1 or 2). Accordingly, the corresponding variable x_i brings, by ϕ, the same truth value to the n clauses of F that contain it. Finally, in G', every clause vertex c_j is incident to two edges labelled 1 and one edge labelled 2. This corresponds to the clause C_j being regarded as satisfied by ϕ only when it has exactly one true variable. □

4.2.2 A positive result for graphs with bounded treewidth

In this section we provide a dynamic programming algorithm that decides if $\mathrm{mE}_k(G) \leq s$ (for given integers k and s) in polynomial time, where G is a graph of bounded treewidth. Apart from the basic notions and notations explained in Section 2.2.2, we need to introduce some additional terminology before proceeding to the main result of this section.

Let $(T, \mathcal{X})$ be a rooted tree-decomposition (with root r) of G and $t \in V(T)$. A *quasi k-labelling* of G_t consists of a pair of functions (ℓ, c), with $\ell : E(G_t) \to \{1, \dots, k\}$ and $c : V_t \to \mathbb{N}$, such that c is a proper vertex-colouring of G_t, for every $v \in V_t \setminus X_t$ we have $c(v) = c_\ell(v)$, and for every $v \in V(X_t)$ we have $c(v) \geq c_\ell(v)$. Intuitively, the notion of quasi k-labelling is a generalisation of proper k-labellings that allows us to further modify the labels (and thus the induced colours) of the edges of X_t if this is needed in order to extend a proper k-labelling of G_t into a proper k-labelling of $G_{t'}$, where t' is the parent of t in T. Finally, let $s_t(\ell) = \sum_{e \in E(G_t)} \ell(e)$.

Observe that every proper k-labelling ℓ' of G induces a quasi k-labelling of G_t. For every $e \in E(G_t)$ and $v \in V_t$, let $\ell(e) = \ell'(e)$ and $c(v) = c_{\ell'}(v)$. The pair (ℓ, c) is a quasi k-labelling of G_t. Indeed, since $(T, \mathcal{X})$ is a tree-decomposition of G, for every internal node t of T, X_t is a separator between $G_t - X_t$ and $G - V_t$. Put differently, there are no edges between vertices of $G_t - X_t$ and $G - V_t$. Furthermore, if r is such that $X_r = \emptyset$, then a quasi k-labelling of G_r is a proper k-labelling of G. Indeed, it is true (by definition) that a quasi k-labelling of G_r differs from a proper k-labelling only on the vertices of X_r and since $X_r = \emptyset$ and $G_r = G$ the observation follows.

Theorem 4.2.5. *Let $k \geq 2$ and $\mathrm{tw} \geq 1$ be two fixed integers. Given a nice graph $G = (V, E)$ with $|V| = n$ and an integer s, the problem of deciding whether $\mathrm{mE}_k(G) \leq s$ holds can be solved in polynomial time if G belongs to the family of graphs that have width at most tw (and in linear time if G is additionally of bounded maximum degree).*

Proof. Let us start by giving some definitions. Let $\Delta = \Delta(G)$ denote the maximum degree of G. For every $t \in V(T)$, let $|X_t| = w_t$, $|E(G[X_t])| = q_t$, $X_t = \{v_1, \dots, v_{w_t}\}$ and $E(G[X_t]) = \{e_1, \dots, e_{q_t}\}$ (to simplify the notation, we will simply denote q_t and w_t by q and w respectively).

Let $\mathcal{F}_t = \{1, \ldots, k\}^q \times \{1, \ldots, k\Delta\}^w \times \{0, \ldots, k\Delta\}^w$ and $(L, FC, CB) \in \mathcal{F}_t$, where $L = \{l_1, \ldots, l_q\}$, $FC = \{f_1, \ldots, f_w\}$ and $CB = \{b_1, \ldots, b_w\}$. The labels we "intent" to assign to the edges of X_t are in L, the "final colours" induced by these labels on the vertices of X_t are in FC, and in CB we can find the contribution to these colours that come "from below" (meaning the part of these final colours that is due to edges between X_t and $G_t - X_t$). Furthermore, for $X'_t \subseteq X_t$ with $X'_t = \{u_1, \ldots, u_{w'}\}$ (where $w' \leq w$), let $FC|_{X'_t} = \{f'_1, \ldots, f'_{w'}\}$ be defined by setting, for each $j \in \{1, \ldots, w'\}$, $f'_j = f_{i_j}$ where $u_j = v_{i_j}$ ($L|_{X'_t}$ and $CB|_{X'_t}$ are defined similarly).

Moreover, a quasi labelling (ℓ, c) of G_t is said *compatible* with $(L, FC, CB) \in \mathcal{F}_t$ if, for each $i \in \{1, \ldots, q\}$ and $j \in \{1, \ldots, w\}$, we have that $\ell(e_i) = l_i, c(v_j) = f_j$, and $b_j = \sum_{x \in N_t(v_j) \setminus X_t} \ell(v_j x)$. This implies that, for all $i \in \{1, \ldots, q\}$ and $j \in \{1, \ldots, w\}$,

$$\sum_{z \in N_t(v_j)} \ell(zv_j) = b_j + \sum_{z \in N(v_j) \cap X_t, e_i = v_j z} l_i \leq c(v_j) = f_j.$$

Among all quasi labellings compatible with (L, FC, CB), let us denote as (ℓ^*, c^*) one such compatible labelling that minimises the sum of the labels assigned by ℓ^*. That is, for any $t \in V(T)$, we have $s_t(\ell^*) \leq s_t(\ell)$ for every quasi labelling ℓ compatible with (L, FC, CB). Let $\alpha_t(L, FC, CB) = s_t(\ell^*)$. In essence, for each possible $(L, FC, CB) \in \mathcal{F}_t$, we have that $\alpha_t(L, FC, CB)$ is equal to the sum of the labels of an optimal (in terms of sum of labels) quasi labelling (ℓ^*, c^*) of G_t that is compatible with (L, FC, CB). Note that not all $(L, FC, CB) \in \mathcal{F}_t$ admit compatible quasi labellings. If $(L, FC, CB) \in \mathcal{F}_t$ has no compatible quasi labelling, then we set $\alpha_t(L, FC, CB) = \infty$.

Finally, let us set

$$\mathrm{Table}(t) = ((L, FC, CB, \alpha_t(L, FC, CB)))_{(L, FC, CB) \in \mathcal{F}_t}$$

being the table associated with each $t \in V(T)$. Note that

$$|\mathrm{Table}(t)| = \mathcal{O}\left(k^q (k\Delta + 1)^{2w}\right) = \mathcal{O}\left(k^{(\mathrm{tw}(G)+1)^2} (k\Delta + 1)^{2\mathrm{tw}(G)+2}\right),$$

since $q \leq \binom{\mathrm{tw}(G)+1}{2} = \mathcal{O}((\mathrm{tw}(G) + 1)^2)$ and $w \leq \mathrm{tw}(G) + 1$. Furthermore, since r is such that $X_r = \emptyset$, then $\alpha_r = \alpha_r(\emptyset, \emptyset, \emptyset)$ is equal to the sum of an optimal proper k-labelling of G_r and thus $\mathrm{Table}(r) = ((\emptyset, \emptyset, \emptyset, \alpha_r))$, where $\alpha_r = \mathrm{mE}_k(G)$. All that remains to be done is to compute this $\mathrm{Table}(t)$ for every $t \in V(T)$. We present a dynamic programming algorithm that performs this computation bottoms up; that is, starting from the leaves of T and progressing towards r. The computation depends on the type of t.

Let t be a leaf node. Recall that $|X_t| = 1$ and thus there are no edges in X_t. For every $y \in \{1, \ldots, k\Delta\}$ and $(L, FC, CB) \in \mathcal{F}_t$, the $\alpha_t(L, FC, CB)$ entry of $\mathrm{Table}(t)$ is defined as:

$$\alpha_t(L, FC, CB) = \begin{cases} 0, & \text{if } (L, FC, CB) = (\emptyset, \{y\}, \{0\}); \\ \infty, & \text{otherwise.} \end{cases}$$

Let t be an introduce node and t' be its unique child. Set $w = |X_{t'}|$, and let v be such that $X_t = X_{t'} \cup \{v\} = \{v_1, \ldots, v_w, v_{w+1} = v\}$. Moreover, let $E(X_t) = \{e_1, \ldots, e_q, e_{q+1}, \ldots, e_{q+h}\}$, where $E(X_{t'}) = \{e_1, \ldots, e_q\}$. Essentially, the set $\{e_{q+1}, \ldots, e_{q+h}\}$ contains the edges between v

and the other vertices of X_t. By induction, we can assume that $\mathrm{Table}(t')$ is already computed. Let us show how to compute $\mathrm{Table}(t)$. Let

$$(L = (l_1, \dots, l_{q+h}), FC = (f_1, \dots, f_{w+1}), CB = (b_1, \dots, b_{w+1})) \in \mathcal{F}_t.$$

There is a quasi labelling of G_t compatible with (L, FC, CB) only if the following three (easily computable) conditions are satisfied:

— The final colour f_{w+1} that corresponds to v is not in conflict with the final colours that correspond to the neighbours of v in X_t. That is, for each $j \in \{1, \dots, w\}$ such that $v_j \in N_t(v) \cap X_t$, we have $f_{w+1} \neq f_j$.

— Since v is introduced in X_t, we have $N_t(v) \setminus X_t = \emptyset$ and, as a consequence, $c(v)$ cannot have any contribution coming from $G_t \setminus X_t$. That is $b_{w+1} = 0$.

— The colour of each vertex in G_t cannot exceed the final colour that corresponds to it. That is, for every $v_i \in X_t$, we must have

$$b_i + \sum_{j=1}^{q+h} l_j \mathbb{I}_{i,j} \leq f_i,$$

where $\mathbb{I}_{i,j} = \begin{cases} 1, & \text{if there is } w \in N(v_i) \cap X_t \text{ such that } wv_i = e_j; \\ 0, & \text{otherwise.} \end{cases}$

If one of these three conditions is not satisfied, then $\alpha_t(L, FC, CB) = \infty$. Otherwise, let us set

$$\alpha_t(L, FC, CB) = \alpha_{t'}(L|_{X_{t'}}, FC|_{X_{t'}}, CB|_{X_{t'}}) + \sum_{j=q+1}^{q+h} l_j.$$

Following the above process, the element $((L, FC, CB), \alpha_t(L, FC, CB))$ is added to $\mathrm{Table}(t)$ for every $(L, FC, CB) \in \mathcal{F}_t$.

Let t be a forget node and t' be its unique child. Set $w = |X_t|$, and let v be such that $X_{t'} = X_t \cup \{v\} = \{v_1, \dots, v_w, v_{w+1} = v\}$. Moreover, let $E(X_{t'}) = \{e_1, \dots, e_q, e_{q+1}, \dots, e_{q+h}\}$, where $E(X_t) = \{e_1, \dots, e_q\}$. By induction, we can assume that $\mathrm{Table}(t')$ is already computed. Let us show how to compute $\mathrm{Table}(t)$.

Let $(L, FC, CB) \in \mathcal{F}_t$. Let $\Gamma_{t'}$ be the subset of $\mathcal{F}_{t'}$ that consists of all $(L', FC', CB') \in \mathcal{F}_{t'}$ such that $(L, FC, CB) = (L'|_{X_t}, FC'|_{X_t}, CB'|_{X_t})$ (*i.e.*, (L, FC, CB) must be the restriction to X_t of some $(L', FC', CB') \in \mathcal{F}_{t'}$) and such that $f'_{w+1} = b'_{w+1} + \sum_{q+1 \leq j \leq q+h} l'_j$. The latter condition allows to respect the property of quasi labellings. Since $v \in G_t - X_t$, the "colour" that it received so far (with a contribution of b'_{w+1} from the vertices in $G_{t'} - X_{t'}$) plus the labels $l'_{q+1}, \dots, l'_{q+h}$ of its incident edges in $X_{t'}$ must equal its "final" colour f'_{w+1}. Finally, let

$$\alpha_t(L, FC, CB) = \min_{\gamma \in \Gamma_{t'}} \alpha_{t'}(\gamma).$$

Following the above process, the element $((L, FC, CB), \alpha_t(L, FC, CB))$ is added to $\mathrm{Table}(t)$ for every $(L, FC, CB) \in \mathcal{F}_t$.

Let t be a join node, t' and t'' be its two children, with $X_t = X_{t'} = X_{t''} = \{v_1, \dots, v_w\}$ and $E(X_t) = E(X_{t'}) = E(X_{t''}) = \{e_1, \dots, e_q\}$. By induction, we can assume that $\mathrm{Table}(t')$ and $\mathrm{Table}(t'')$ have already been computed. Let us show how to compute $\mathrm{Table}(t)$.

Let $(L, FC, CB) \in \mathcal{F}_t$. Let Γ_t be the set of pairs $((L, FC, CB'), (L, FC, CB''))$ such that $(L, FC, CB') \in \mathcal{F}_{t'}$ and $(L, FC, CB'') \in \mathcal{F}_{t''}$ such that $CB = CB' + CB''$ (meaning that for each $j \in \{1, \ldots, w\}$ we have $b_j = b'_j + b''_j$, where $b_j \in CB$, $b'_j \in CB'$ and $b''_j \in CB''$). Then, let

$$\alpha_t(L, FC, CB) = \min_{((L,FC,CB'),(L,FC,CB''))\in\Gamma_t} \alpha_{t'}(L, FC, CB') + \alpha_{t''}(L, FC, CB'') - \sum_{i=1}^{q} l_i.$$

Following the above process, the element $((L, FC, CB), \alpha_t(L, FC, CB))$ is added to $\mathrm{Table}(t)$ for every $(L, FC, CB) \in \mathcal{F}_t$.

In all cases, it can be shown by induction that $\alpha_t(L, FC, CB) \neq \infty$ if and only if there is a quasi labelling of G_t compatible with (L, FC, CB), and, moreover, that if $\alpha_t(L, FC, CB) \neq \infty$, then it is the minimum sum of the edge labels among all quasi labellings of G_t compatible with (L, FC, CB). □

An alternative way to interpret Theorem 4.2.5 is that, given a graph G and two integers k and s, the problem of deciding if $\mathrm{mE}_k(G) \leq s$ is in FPT when considering as a parameter the treewidth plus the maximum degree of G. It is worth mentioning here that in Chapter 6 we will demonstrate a similar result for the problem investigated in that chapter.

4.3 Particular behaviours of the problem

In this section, we study some behaviours of the problem of determining $\mathrm{mE}_k(G)$ for some integer k and nice graph G. We start by establishing that there is no systematic relationship between the proper labellings we are interested in and those considered in [16] and [31], where the authors study proper labellings with a minimum number of induced colours or that minimise the maximum induced colour respectively. In particular, in Section 4.3.1, we show that a labelling of a graph that minimises the maximum induced colour, does not necessarily also minimise the sum of assigned labels, and *vice versa*. Then, in Section 4.3.2, we prove that, in general, using large labels (larger than $\chi_\Sigma(G)$) might be needed for designing proper labellings ℓ verifying $\sigma(\ell) = \mathrm{mE}(G)$. This actually remains true in cases where G is a tree.

4.3.1 Minimising the maximum colour versus minimising the sum of labels

As mentioned already, quite recently the authors of [31] investigated proper labellings that minimise the maximum resulting vertex colour. The formal definitions are as follows. For a given graph G and a labelling ℓ of G, let $\mathrm{mS}(G, \ell)$ denote the maximum vertex colour $c_\ell(v)$ induced by ℓ over all vertices v of G. For a given $k \geq \chi_\Sigma(G)$, let $\mathrm{mS}_k(G)$ denote the smallest value of $\mathrm{mS}(G, \ell)$ over all proper k-labellings of G. Now, the main parameter of interest is $\mathrm{mS}(G)$, which is defined as the minimum value of $\mathrm{mS}_k(G)$ over all values of $k \geq \chi_\Sigma(G)$. The authors of [31] establish that for any fixed $k \geq 2$, calculating $\mathrm{mS}_k(G)$ is NP-hard, even when G is a bipartite graph. They also provide a polynomial-time algorithm for solving this problem on graphs of bounded treewidth. Then they propose upper and lower bounds on the investigated parameter for bipartite graphs, and show that if G is a tree of maximum degree Δ, then $\mathrm{mS}_k(G) \in \{\Delta, \Delta+1, \Delta+2\}$, for any $k \geq 2$. Finally, they show that using larger labels can actually be beneficial when constructing labellings that minimise the maximum induced colour. A similar behaviour is also exhibited by the problem investigated in this chapter in Section 4.3.2.

As established in Observation 4.1.3 determining $\mathrm{mE}(G)$ for a nice graph G can equivalently be seen as finding a proper labelling of G that minimises the sum of resulting vertex colours. Thus, one could think that maybe $\mathrm{mE}(G)$ is a good approximation of $\mathrm{mS}(G)$, or *vice versa*. In this section, we show that this is actually not the case.

The next result shows that, when constructing a proper labelling ℓ of a graph G with $\mathrm{mS}(G,\ell) = \mathrm{mS}(G)$, we might have $\sigma(\ell)$ being arbitrarily far from $\mathrm{mE}(G)$. In other words, minimising the maximum colour does not imply minimising the sum of labels. This actually remains true for trees.

Theorem 4.3.1. *There exist nice trees T with arbitrarily large maximum degree $\Delta \geq 2$ for which, for any proper labelling ℓ achieving $\mathrm{mS}(T,\ell) = \mathrm{mS}(T)$, we have $\mathrm{mE}(T,\ell) = \mathrm{mE}(T) + \Delta - 2$.*

Proof. Consider the following tree T with maximum degree $\Delta \geq 2$. We start from a vertex v with Δ neighbours $u_1, \dots, u_\Delta$, each of which is adjacent to $\Delta - 1$ leaves. In other words, all neighbours of v have degree Δ, and all other vertices are leaves at distance exactly 2 from v.

Now consider a proper labelling ℓ of T that minimises the maximum colour, *i.e.*, $\mathrm{mS}(T,\ell) = \mathrm{mS}(T)$. Since T has adjacent vertices with degree Δ, we have $\mathrm{mS}(T,\ell) \geq \Delta + 1$. One possible way to attain $\mathrm{mS}(T,\ell) = \Delta + 1$ is to have all edges incident to v being labelled 1, and, for each u_i, to have exactly one incident edge going to a leaf being labelled 2 and all other $\Delta - 2$ incident edges being labelled 1. Indeed, we get $c_\ell(v) = \Delta \neq \Delta + 1 = c_\ell(u_i)$ for every $i \in \{1, \dots, \Delta\}$. Actually, this is the only way to have $\mathrm{mS}(T,\ell) = \Delta + 1$, because if we label the edges incident to v so that $c_\ell(v) = \Delta + 1$, then it is easy to see that the vertex u_i such that $\ell(vu_i) = 2$ would get $c_\ell(u_i) \geq \Delta + 2$ to avoid a colour conflict between v and u_i. Therefore, there is only one general way to label (actually 2-label) T so that $\mathrm{mS}(T,\ell) = \mathrm{mS}(T) = \Delta + 1$, and we note that the number of edges labelled 2 by ℓ is exactly Δ (one for each u_i). Thus $\mathrm{mE}(T,\ell) = |E| + \Delta$.

Observe now that regardless of the value of Δ, the 2-labelling ℓ^* of T where $\ell^*(vu_1) = \ell^*(vu_2) = 2$ and all other edges are labelled 1 is proper. This is because we get $c_{\ell^*}(v) = \Delta + 2$, $c_{\ell^*}(u_1) = c_{\ell^*}(u_2) = \Delta + 1$ and $c_{\ell^*}(u_i) = \Delta$, for $i \in \{3, \dots, \Delta\}$. Thus, $\mathrm{mE}(T) \leq \mathrm{mE}_2(T) \leq \mathrm{mE}(T,\ell^*) = |E| + 2$ and the difference between $\sigma(\ell)$ and $\sigma(\ell^*)$ then gets arbitrarily large as Δ grows larger. □

The next result shows that the converse is also true: a proper labelling that minimises the sum of labels does not necessarily minimise the maximum colour as well. Note that in what follows, we make use of the $G^{\curlywedge}$ gadget, introduced in Section 2.3.2, Definition 2.3.3 and illustrated in Figure 2.5.

Theorem 4.3.2. *There exist nice graphs G with arbitrarily large maximum degree $\Delta \geq 12$ for which, for any proper 2-labelling ℓ achieving $\sigma(\ell) = \mathrm{mE}_2(G)$, we have $\mathrm{mS}(G,\ell) = \mathrm{mS}(G) + \Delta$.*

Proof. Having a closer look at $G^{\curlywedge}$, we note that, regarding the problem of computing $\mathrm{mE}(G^{\curlywedge})$, it has the following property:

Claim 4.3.3. *Let ℓ be a proper 2-labelling of $G^{\curlywedge}$ achieving $\sigma(\ell) = \mathrm{mE}_2(G^{\curlywedge})$. Then ℓ assigns label 2 to the input and two outputs of $G^{\curlywedge}$.*

Proof of the claim. As mentioned in Section 2.3.2, the proper 2-labellings of $G^{\curlywedge}$ are of two kinds: those in L_1 assigning label 1 to the input and two outputs, and those L_2 assigning label 2 to the input and two outputs. Such groups of labellings L_1 and L_2 are as described in Figure 2.5 (a) and (b), respectively. In particular, it is important to recall that the solid edges in the figures must be labelled as illustrated (up to symmetry), while the only sources of freedom we have are the labels assigned to the wiggly edges, which can each freely be chosen to be 1 or 2 (recall Theorems 2.3.4 and 2.3.5).

Let us now determine the minimum sum of labels assigned by these proper labellings:

— For a proper 2-labelling $\ell_1 \in L_1$ of $G^{\curlywedge}$ assigning label 1 to the input and two outputs, we note, as illustrated in Figure 2.5 (a), that u_5 and u_6 must get colour 10, which is possible only if the two copies of T_2 attached to u_5 and u_6 are labelled as in Figure 2.4 (a). Indeed, assuming that one of these copies of T_2, say the one attached to u_5, is labelled as in Figure 2.4 (b), would mean that $c_{\ell_1}(u_5) = c_{\ell_1}(u_3) = 10$, and thus ℓ_1 would not be a proper labelling. The vertex u_7 must get colour 11, which is not prevented by any of the two ways of labelling the copy of T_2 attached to it. Thus, for ℓ_1 to minimise the sum of labels, the copies of T_2 attached to u_5 and u_6 must be labelled as in Figure 2.4 (a), and the copy of T_2 attached to u_7 must be labelled as in Figure 2.4 (a) as well (as the sum of labels in the labelling of Figure 2.4 (b) is larger). All wiggly edges should be assigned label 1. In total, a minimum (in regards to the sum of labels) labelling $\ell_1 \in L_1$ assigns label 1 to 40 edges and label 2 to 25 edges, and thus $\sigma(\ell_1) = 90$.

— By similar arguments, we deduce that, by a proper 2-labelling $\ell_2 \in L_2$ of $G^{\curlywedge}$ assigning label 2 to the input and two outputs, the copies of T_2 attached to u_5 and u_6 should be labelled as depicted in Figure 2.4 (b), while the copy of T_2 attached to u_7 should be labelled as depicted in Figure 2.4 (a) (in particular, the two labellings of T_2 depicted in Figure 2.4 comply with u_7 having colour 7). Again, the wiggly edges should be labelled 1. In total, a minimum labelling $\ell_2 \in L_2$ assigns label 1 to 42 edges and label 2 to 23 edges, and thus $\sigma(\ell_2) = 88$.

Thus, a proper 2-labelling ℓ of $G^{\curlywedge}$ achieving $\sigma(\ell) = \mathrm{mE}_2(G^{\curlywedge})$ must assign label 2 to the input and two outputs. ◇

Recall now the graph G_m, resulting from attaching $m-1 \geq 3$ copies of the graph $G^{\curlywedge}$ (described in Section 2.3.2, Definition 2.3.6). Recall also that, according to Theorem 2.3.7, by any proper labelling ℓ of G_m, the input and the m outputs of G_m will be assigned the same label. Let L_1 be the labellings assigning label 1 to the input and all outputs, and L_2 be the labellings assigning label 2 to the input and all outputs of G_m. Clearly, a proper 2-labelling ℓ of G_m verifying $\sigma(\ell) = \mathrm{mE}_2(G_m)$, when restricted to any constituting copy of $G^{\curlywedge}$ in G_m, should also be minimum in terms of sum of assigned labels. From Claim 4.3.3, we thus deduce that a proper 2-labelling ℓ of G_m achieving $\sigma(\ell) = \mathrm{mE}_2(G_m)$ must assign label 2 to the input and all outputs, *i.e.*, must belong to the L_2 group mentioned above. In particular, the difference between the sum of labels of a minimum $\ell_1 \in L_1$ and the sum of labels of a minimum $\ell_2 \in L_2$ gets larger as the number of copies involved in the construction of G_m gets larger.

Now let G be the graph obtained as follows. Consider the graph G_Δ with Δ outputs (for any $\Delta \geq 12$), and identify the degree-1 vertices of these Δ outputs to a single vertex o^* (with maximum degree Δ, as it can be checked from Figures 2.4 and 2.5 that all other vertices of G_m have degree at most 6). Note that a proper 2-labelling of G is also proper for G_Δ, since vertices of degree 1 cannot be involved in conflicts. Also, a proper 2-labelling of G_Δ must be proper in G as well, since o^* has degree at least 12 while all its neighbours have degree 2. By these arguments, a proper 2-labelling ℓ of G verifying $\sigma(\ell) = \mathrm{mE}_2(G)$ must thus be one of these proper 2-labellings in L_2 assigning label 2 to the input and all outputs. Such an $\ell_2 \in L_2$ verifies $c_{\ell_2}(o^*) = 2\Delta$. On

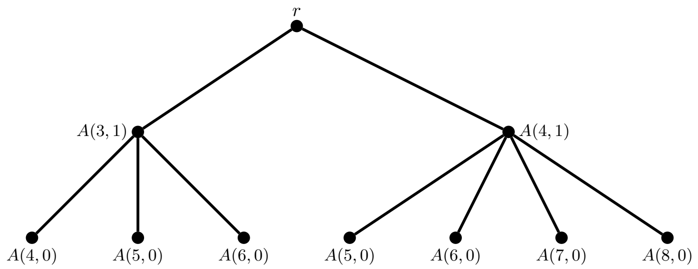


Figure 4.2 – The auxiliary graph $A(2,2)$.

the other hand, a proper 2-labelling $\ell_1 \in L_1$ assigning label 1 to all outputs verifies $c_{\ell_1}(o^*) = \Delta$. Thus, a proper 2-labelling ℓ of G verifying $\sigma(\ell) = \mathrm{mE}_2(G)$ will make o^* get colour 2Δ, while there are proper 2-labellings by which o^* gets colour Δ. Note in particular that by our choice of Δ, vertex o^* must indeed be the vertex with the largest colour, as its degree is at least 12, all other vertices have degree at most 6, and we are only assigning labels 1 and 2. $\square$

4.3.2 Using larger labels can be arbitrarily better

In this section, we present, for any $k \geq 3$, a construction for a tree T_k such that $\mathrm{mE}_2(T_k) = \mathrm{mE}_3(T_k) = \mathrm{mE}_4(T_k) = \cdots = \mathrm{mE}_k(T_k)$ and $\mathrm{mE}_{k+1}(T_k) < \mathrm{mE}_k(T_k)$. In other words, for these trees T_k we need to consider larger labels to design a proper labelling ℓ achieving $\mathrm{mE}(T_k, \ell) = \mathrm{mE}(T_k)$.

Let us first introduce the *auxiliary graph* $A(\alpha, \beta)$ (for $\alpha \geq 2$ and $\beta \geq 0$), which will serve as the building block for T_k. This auxiliary graph is a tree and is built recursively as follows: for any $\alpha^* \in \mathbb{N}$, define $A(\alpha^*, 0)$ as a leaf. For any $\beta > 0$, $A(\alpha, \beta)$ is a tree of height β, rooted in a vertex r that has α children. For each $1 \leq i \leq \alpha$, let c_i be the corresponding child of r; each c_i is the root of an $A(\alpha+i, \beta-1)$ tree and thus $d(c_i) = \alpha+i+1$ (since each c_i has $\alpha+i$ children of its own as well as an edge connecting him with his parent). Note that $d(c_i) \in D(\alpha) = \{\alpha+2, \dots, 2\alpha+1\}$ and that, for $i \neq j$, we have $d(c_i) \neq d(c_j)$ (and thus all the values of $D(\alpha)$ are used exactly once). Finally, we say that $A(\alpha, \beta)$ is *represented* by r. The auxiliary graph $A(2,2)$ is illustrated in Figure 4.2.

Let us also define the *pending auxiliary graph* that corresponds to $A(\alpha, \beta)$ as $P(\alpha, \beta) = (V, E)$, where $V = V(A(\alpha, \beta)) \cup \{v\}$ and $E = E(A(\alpha, \beta)) \cup \{vr\}$; in essence $P(\alpha, \beta)$ is $A(\alpha, \beta)$ with an extra vertex v connected to r. We say that $P(\alpha, \beta)$ is *pending from* v. Observe that $P(\alpha, \beta)$ is locally irregular and thus the labelling ℓ that assigns label 1 to every one of its edges is proper and verifies $\mathrm{mE}(P(\alpha, \beta), \ell) = |E|$.

Lemma 4.3.4. *Let $\beta \in \mathbb{N}^*$ and $\alpha \geq 2$. Let ℓ be a proper α-labelling of the pending auxiliary graph $P(\alpha, 2\beta)$ pending from v. Let $u, w \in V(P(\alpha, 2\beta))$ such that $1 \leq \mathrm{dist}(u, v) \leq 2$ and w is the parent of* u*. If $\ell(uw) > 1$, then $\mathrm{mE}(P(\alpha, 2\beta), \ell) \geq |E| + \beta$.*

Proof. Let us prove the claim for the case where $\mathrm{dist}(u, v) = 2$ and u is the root of an $A(\alpha+1, 2\beta-1)$ subtree and $w = r$ (similar arguments hold for the other cases) and let us first assume that uw is the only edge of $P(\alpha, \beta)$ that has label more than 1, say $\ell(uw) = \alpha'$ where $2 \leq \alpha' \leq \alpha$. It follows that $c_\ell(u) = \alpha + \alpha' + 1$ and that $\alpha + 3 \leq c_\ell(u) \leq 2\alpha + 1$. Since all edges of $P(\alpha, 2\beta)$ except uw are labelled 1, each child y of u has $c_\ell(y) = d(y)$. Moreover, since u is the root of the

$A(\alpha+1, 2\beta-1)$ tree, each one of the $\alpha+1$ children of u has a unique degree in the set $D(\alpha+1)$. But $D(\alpha+1) = \{\alpha+3, \dots, 2\alpha+2\}$ and $c_\ell(u) \in D(\alpha+1)$. It follows that there exists a child of u that has, by ℓ, the same colour as u. Thus ℓ must assign a label different than 1 to at least one more edge of $P(\alpha, \beta)$, and the argument can be repeated at least β times (since the height of $T(\alpha, 2\beta)$ is $2\beta+1$), leading to ℓ having to assign a label different than 1 to at least β edges. The exact value of $\mathrm{mE}(P(\alpha, 2\beta), \ell) = |E| + \beta$ is reached if each time the argument is repeated, $\alpha' = 2$ and the next edge that gets assigned label 2 is at distance 2 from the previous ones. □

Theorem 4.3.5. *For every $k \geq 2$, there exists a nice graph T_k such that $\mathrm{mE}_{k+1}(T_k) < \mathrm{mE}_k(T_k)$.*

Proof. Let $k \geq 2$ and let us describe the construction of $T_k = (V, E)$. For $0 \leq j \leq k-1$, let $P(k+j, 2(k+1))$ be the auxiliary graph pending from v_j that corresponds to an auxiliary graph $A(k+j, 2(k+1))$ (represented by a vertex r_j) and let u, v be two vertices connected by the edge uv. The tree T_k is the graph that is produced by identifying v with each one of the v_j. Observe that since r_j represents $A(k+j, 2(k+1))$, each r_j has $d(r_j) = k+j+1$ in T_k and that the height of T_k is $2(k+1)+1$. Also observe that in T_k, since $N(v) = \{r_0, \dots, r_{k-1}, u\}$, we have $d(v) = k+1 = d(r_0)$.

Claim 4.3.6. *There exists a proper $(k+1)$-labelling ℓ of T_k such that $\sigma(\ell) = |E| + k$.*

Proof of the claim. Note that T_k is almost locally irregular. Indeed, let w be a non-leaf vertex of T_k different from r_1, v and u, and let x be its parent. If $d(w) = d+1$, then $d+1 > d(x)$ (by construction) and w has d children, each one having degree at least $d+2$. In fact, the only adjacent vertices that have the same degree are v and r_0.

Let ℓ be the $(k+1)$-labelling of T_k that assigns label $k+1$ to the edge uv and label 1 to the remaining edges of T_k. Then $c_\ell(v) = 2k+1$ and, for each $0 \leq i \leq k-1$, we have $d(r_j) \in [k+1, 2k]$ and thus there is no conflict between the colour of v and that of its children. It follows that ℓ is a proper $(k+1)$-labelling for T_k and $\sigma(\ell) = |E| + k$. ◇

Let ℓ' be any proper k-labelling of T_k. It suffices to show that $\sigma(\ell') > |E|+k$. Note that, since $d(v) = d(r_0) = k+1$ and ℓ' is proper, there must exist vertices w, y, with $w \in N(r_0) \setminus \{v\}$ and $y \in N(v) \setminus \{u, r_0\}$, such that at least one of the edges uv, r_0w or vy has to have a label different from 1. Let $\ell'(uv) = l$ with $2 \leq l \leq k$ and assume that this is the only edge of T_k that has a label different from 1. Then $c_{\ell'}(v) = k+l$ and $k+l \in \{k+2, \dots, 2k\}$. Recall that for every $0 \leq j \leq k-1$, vertex r_j has $d(r_j) = k+j+1$ and thus $d(r_j) \in \{k+1, \dots, 2k\}$. Besides, since uv is the only edge with a label different from 1, we have $c_{\ell'}(r_j) = d(r_j)$. It follows that there exists a $j \in \{0, \dots, k-1\}$, such that $c_{\ell'}(r_j) = c_{\ell'}(v)$ leading to ℓ' not being proper. Thus, there must exist another edge $u'v'$ (with, say, u' being the parent of v') that is assigned a label different from 1 by ℓ'. Note that u' is either r_j or v. This edge, however, belongs to $P(q, 2(k+1))$ (for some $q \in \{k, \dots, 2k-1\}$) and we have that $1 \leq \mathrm{dist}(v', v) \leq 2$. It follows from Lemma 4.3.4 that $\mathrm{mE}(T_k) \geq |E| + k + 1$. The cases where r_0w or vy are assigned a label different from 1 follow by applying directly Lemma 4.3.4. □

Observe that the height of T_k can be controlled by changing the value β of the pending auxiliary graphs that form it. Furthermore, for $\alpha \geq 2$ and $\beta, \beta' \in \mathbb{N}^*$ with $\beta < \beta'$, it follows from Lemma 4.3.4 that $\mathrm{mE}(T(\alpha, 2\beta)) < \mathrm{mE}(T(\alpha, 2\beta'))$, where $T(\alpha, 2\beta)$ denotes the graph T_k formed by α pending auxiliary graphs of height 2β. This proves the following corollary:

Corollary 4.3.7. *For every $k \geq 2$, there exists a graph T_k such that $\mathrm{mE}_{k+1}(T_k)$ is arbitrarily smaller than $\mathrm{mE}_k(T_k)$.*

4.4 Bounds

Observation 4.1.1 establishes that, for any nice graph G, in general $\mathrm{mE}(G)$ should be expressed as a function of $|E(G)|$. To date, the best result towards the 1-2-3 Conjecture, due to Kalkowski, Karoński and Pfender [79], states that $\chi_\Sigma(G) \leq 5$ holds for every nice graph G. This implies Theorem 4.1.2, stating that for every nice graph G, $\mathrm{mE}(G) \leq \mathrm{mE}_5(G) \leq 5|E(G)|$.

Throughout this section, we provide results towards Conjecture 2.1.4, which claims that for every nice graph G, $\mathrm{mE}(G) \leq 2|E(G)|$. In particular, we prove a weaker version of that conjecture for graphs with given chromatic number, we verify Conjecture 2.1.4 for bipartite graphs, and we prove a stronger result in the particular case of trees.

4.4.1 Graphs with large chromatic number

Towards Conjecture 2.1.4, we provide a general upper bound on $\mathrm{mE}(G)$ being a function of the chromatic number $\chi(G)$. In particular, the bound we get is better than that in Theorem 4.1.2, and even better than the conjectured one in Conjecture 2.1.4, for dense enough graphs. In the upcoming proofs, we make use of arguments that have already been presented in Section 2.2.1.

The next results are for graphs that are not bipartite. Results dedicated to bipartite graphs will be provided in the next section.

Theorem 4.4.1. *Let $G = (V, E)$ be a nice connected graph with chromatic number $k = \chi(G)$ at least 3. Then, we have $\mathrm{mE}(G) \leq \mathrm{mE}_{k+1}(G) \leq |E(G)| + k|V(G)|$.*

Proof. Let H be an odd unicyclic spanning subgraph of G (as defined in Section 2.2.1). Also, let $G' = G \cup S_0$, where $S_0 = \emptyset$, and $S_i \subseteq V(G')$, for $1 \leq i \leq k$, be the k stable sets induced by a proper vertex-colouring c of G' (*i.e.*, if $v \in S_i$ then $c(v) = i$). We are going to construct a $(k+1)$-labelling ℓ on the edges of G' such that $\sigma(\ell) \leq |E(G')| + k|V(G')|$. Let us start by having ℓ assigning label 1 to all edges of G'. At this point, the colour of every vertex is exactly its degree. For each $0 \leq i \leq k$, let $S_i^* = \{v \in S_i | c_\ell(v) = i \bmod k+1\}$ (obviously $S_0^* = \emptyset$). Our goal is to modify ℓ so that for each i, we have $S_i^* = S_i$, from which it follows that c_ℓ is a proper vertex-colouring of G'. Aiming at reaching that conclusion, note that, modulo $k+1$, we can equivalently have ℓ assigning labels $0, \dots, k$ instead.

Let $v^* \in V(G')$ such that $d(v) = d \bmod k+1$. Since $c_\ell(v^*) = d(v^*)$, it follows that $v^* \in S_d^*$. Free to relabel the stable sets induced by c, we may assume that d is such that $S_{d-1} = \emptyset$. For each $v \in S_i \setminus S_i^*$, we define $P_o(v) = (v^*, h_1^o, \dots, h_n^o, v)$ and $P_e(v) = (v^*, h_1^e, \dots, h_m^e, v)$ to be an odd and an even walk, respectively, following the edges of H, that connect v^* and v (thus n is an even number and m is an odd number). These walks are sure to exist because H contains the odd cycle C. We modify ℓ to a labelling ℓ' as follows. We traverse $P_o(v)$ from one end to the other, and, as going along, alternate between removing $1 \bmod k+1$ and adding $1 \bmod k+1$ from the labels of the traversed edges. Thus $\ell'(v^*h_1^o) = \ell(v^*h_1^o) - 1 \bmod k+1, \ell'(h_1^o h_2^o) = \ell(h_1^o h_2^o) + 1 \bmod k+1, \dots, \ell'(h_n^o v) = \ell(h_n^o v) - 1 \bmod k+1$. We perform similar modifications as traversing $P_e(v)$ from one end to the other. Thus $\ell'(v^*h_1^e) = \ell(v^*h_1^e) + 1 \bmod k+1, \ell'(h_1^e h_2^e) = \ell(h_1^e h_2^e) - 1 \bmod k+1, \dots, \ell'(h_m^e v) = \ell(h_m^e v) - 1 \bmod k+1$. These modifications do not affect the colours of the internal vertices of $P_o(v)$ and $P_e(v)$. We perform these modifications one after the other. That is, if we start by modifying $P_o(v)$, then we continue by modifying $P_e(v)$, next with modifying $P_o(v)$, and so on. Each time we modify $P_o(v)$ or $P_e(v)$, the colour of v is reduced by $1 \bmod k+1$ and if we alternate between modifying $P_o(v)$ and $P_e(v)$, then the colour of v^* stays the same.

Let $v \in S_i \setminus S_i^*$. We alternate between modifying $P_o(v)$ and $P_e(v)$ until $c_{\ell'}(v) = i \bmod k+1$. Then me move on to modifying another vertex $v' \in S_{i'} \setminus S_{i'}^*$ (i is not necessarily different from i'). If the last modification for v was on $P_o(v)$, then we start with modifying $P_e(v')$ and once more we proceed by alternating between modification on $P_e(v')$ and $P_o(v')$ and *vice versa*.

It is clear that once the above process is finished, for each $v \neq v^*$ and i, we have $v \in S_i$ if and only if $c_{\ell'}(v) = i \bmod k+1$. Also, if the total number w of modifications done is even, then $c_{\ell'}(v^*) = d \bmod k+1$ and if w is odd, then $c_{\ell'}(v^*) = d-1 \bmod k+1$. In any case, and since before the modifications we had $S_{d-1} = \emptyset$, the vertex-colouring $c_{\ell'}$ of G is proper. Note that this remains true when turning all labels 0 into $k+1$, so that ℓ' is a proper $(k+1)$-labelling as desired. Recall also that the modifications are done on the edges of H and $|E(H)| = |V(G)|$. In the worst case, all the edges of H are labelled $k+1$ by ℓ' and thus $\mathrm{mE}(G, \ell') \leq |E| + k|V|$. □

In some contexts making use of the walk-switching procedure described in the proof of Theorems 4.4.1 and 2.2.1, there are favourable situations in which the bound can be further reduced. The next result illustrates that fact.

Theorem 4.4.2. *Let $G = (V, E)$ be a nice connected graph with odd chromatic number $k = \chi(G)$ at least 3. Then, we have $\mathrm{mE}(G) \leq \mathrm{mE}_k(G) \leq |E| + (k-1)|V|$.*

Proof. Let H be an odd unicyclic spanning subgraph of G and S_i (for $0 \leq i \leq k-1$) be the stable sets induced by a proper k-vertex-colouring c of G. Our goal is to reach, by ℓ, the desired colours modulo k. Under that assumption, we can here assign labels $0, \dots, k-1$ instead. Once more, we start with ℓ assigning label 1 to all edges of G. For each $1 \leq i \leq k$, let $S_i^* = \{v \in S_i \mid c_\ell(v) = i \bmod k\}$.

For each $v \in S_i \setminus S_i^*$, let P_v be an odd-length closed walk of H that contains v. Again the existence of P_v is guaranteed because of C. We proceed by modifying the labels of P_v: we alternate between adding 1 (modulo k) and removing 1 (modulo k) from the labels of consecutive edges of P_v. Since P_v is a closed walk of odd length, exactly two consecutive edges (not necessarily distinct) will have to be altered in the same way (*i.e.*, either they are both incremented by 1 or reduced by 1 modulo k). The modification is done so that these two edges have v as a common vertex. Let ℓ' be the modified ℓ and let us assume that the labels of the edges of H that are adjacent to v are both incremented by 1 modulo k (symmetric arguments hold for the other case). Clearly $c_{\ell'}(v) = c_\ell(v) + 2 \bmod k$ and since k is odd, by repeating this process the desired value $c_{\ell'}(v) = i \bmod k$ is eventually reached.

Eventually turn all 0s into k's. In the worst case, ℓ' assigns label k on each one of the $|V|$ edges of H. Thus $\mathrm{mE}(G, \ell') \leq |E| + (k-1)|V|$. □

4.4.2 Bipartite graphs

In this section, we prove Conjecture 2.1.4 for nice bipartite graphs. It turns out, however, that we are not aware of many connected bipartite graphs G for which $\mathrm{mE}_3(G)$ reaches exactly $2|E(G)|$. To go further, we both improve the upper bound in particular contexts, and exhibit constructions of connected bipartite graphs G with large value of $\mathrm{mE}_2(G)$, that are legitimate candidates for having $\mathrm{mE}(G)$ large. Throughout this section, it is worth keeping in mind that determining $\mathrm{mE}_2(G)$ for a given bipartite graph G is NP-complete by Theorem 4.2.2.

4.4.2.1 Conjecture 2.1.4 for nice bipartite graphs

Recall that any bipartite graph belongs to exactly one of the sets $\mathcal{B}_1$, $\mathcal{B}_2$ or $\mathcal{B}_3$ where, for each $i \in \{1,2,3\}$, $\mathcal{B}_i$ contains exactly the connected bipartite graphs G with $\chi_\Sigma(G) = i$. Note that $\mathcal{B}_1$ consists of the locally irregular bipartite graphs G, each one of which verifies $\mathrm{mE}_1(G) = |E(G)|$. The graphs G of $\mathcal{B}_2$ admit proper 2-labellings, and, for these, by Observation 4.1.1 we have $\mathrm{mE}_2(G) \leq 2|E(G)|$. So, in order to prove Conjecture 2.1.4 for nice bipartite graphs, we only need to focus on the graphs of $\mathcal{B}_3$. These graphs are what we call odd multi-cacti, and are defined in Section 2.3.1.

We are now ready to prove our main result in this section, for which we will make use of Lemma 2.2.2, presented in Chapter 2.

Theorem 4.4.3. *For every nice connected bipartite graph G, we have* $\mathrm{mE}(G) \leq \mathrm{mE}_3(G) \leq 2|E(G)|$.

Proof. Since the statement holds for $G \in \mathcal{B}_1 \cup \mathcal{B}_2$, as explained earlier, we can assume $G \in \mathcal{B}_3$, *i.e.*, G is an odd multi-cactus with bipartition (U, V) (where both $|U|$ and $|V|$ are odd by definition). If G is a cycle with length at least 6 congruent to 2 modulo 4, then the result follows from Theorem 4.1.6. Thus, we may assume that $\Delta(G) \geq 3$, *i.e.*, some path attachments were made to build G starting from an original cycle.

Let us consider the last olive edge xy to which a path $P = (x, v_1, \dots, v_{4k}, y)$ was attached in the construction of G, where $k \geq 1$. Recall that $d(x) = d(y) \geq 3$ by construction. Consider $G' = G - \{v_1, v_2, v_3\}$. Assuming $v_1, v_3 \in U$ and $v_2 \in V$, the bipartition of G' is $(U', V') = (U \setminus \{v_1, v_3\}, V \setminus \{v_2\})$. This means that $|V'|$ is even. By Lemma 2.2.2, there is a proper 2-labelling ℓ' of G' such that all vertices of U' have even colour while all vertices of V' have odd colour. Since $x \in V'$, the colour $c_{\ell'}(x)$ is odd, and thus at least 3 since $d_{G'}(x) \geq 2$. Similarly, $v_4 \in V'$, so the colour $c_{\ell'}(v_4)$ is odd, and it is precisely 1 since $d_{G'}(v_4) = 1$.

We now extend ℓ' to a proper 3-labelling ℓ of G, by assigning label 1 to v_1v_2, label 2 to xv_1 and v_3v_4, and label 3 to v_2v_3. This way:

- $c_\ell(x)$ and $c_\ell(v_4)$ remain odd;
- $c_\ell(v_1) = 3 < 5 \leq c_\ell(x)$;
- $c_\ell(v_3) = 5 > 3 = c_\ell(v_4)$;
- $c_\ell(v_2) = 4 \notin \{c_\ell(v_1), c_\ell(v_3)\} = \{3, 5\}$.

For these reasons, it should be clear that ℓ is indeed proper. We additionally note that label 3 is actually assigned only once by ℓ, to v_2v_3. Furthermore, ℓ assigns label 1 at least once, *e.g.* to v_1v_2. From this, it follows that $\sigma(\ell) \leq 2|E(G)|$. ☐

As mentioned earlier, the only connected bipartite graph G verifying $\mathrm{mE}(G) = 2|E(G)|$ we are aware of, is C_6. Due to the small number of edges of C_6, this case looks quite pathological. In particular, it is natural to wonder whether Theorem 4.4.3 can be improved in general, when excluding C_6. We investigate this concern in what follows.

4.4.2.2 Lower bounds for some bipartite graphs

Our main result in this section is that, in general, for a nice connected bipartite graph G it is not possible to lower $\mathrm{mE}_2(G)$ below the $\frac{3}{2}|E(G)|$ barrier. Put differently, there exist connected bipartite graphs for which label 2 must be assigned to at least half of the edges by any proper 2-labelling. This is a consequence of the following more general result, which is of independent interest.

Theorem 4.4.4. *Let G be any nice connected graph, and let H be a graph obtained from G by subdividing every edge e exactly n_e times, where $n_e = 4k_e+3$ for some $k_e \geq 0$. Then $\chi_\Sigma(H) = 2$. Furthermore,* $\mathrm{mE}_2(H) = \frac{3}{2}|E(H)|$.

Proof. For every edge $e = uv$ of G, let us denote by P_e the corresponding path of length $4(k_e+1)$ in H. Note that H has many adjacent 2-vertices, so $\chi_\Sigma(H) > 1$. Also, H is bipartite with bipartition (X, Y), where w.l.o.g. X contains all vertices of G. Now let ℓ be the 2-labelling of H obtained by considering every edge $e = uv$ of G, and assigning labels $2, 1, 1, 2, 2, 1, 1, \dots, 1, 1, 2$ to the consecutive edges of P_e as going from u to v. Then ℓ is proper since all vertices in X have even colour, while all vertices in Y have odd colour. The last part of the claim follows from the fact that for every edge e of G, in any labelling ℓ of H every two edges of P_e being at distance 2 apart must receive distinct labels (recall Observation 2.1.5). Due to the length of P_e, this implies that the sum of the labels assigned to its edges is at least $\frac{3}{2}|E(P_e)|$. Thus, $\sigma(\ell) \geq \frac{3}{2}|E(H)|$. □

Corollary 4.4.5. *There exist infinitely many connected bipartite graphs $G \in \mathcal{B}_2$ verifying* $\mathrm{mE}_2(G) = \frac{3}{2}|E(G)|$. *This remains true for trees.*

Proof. This follows from Theorem 4.4.4. The last part of the statement is because any subdivision of a tree is clearly a tree itself. □

In particular through experimentation via computer programs, we also managed to come up with the following construction yielding connected bipartite graphs G for which $\mathrm{mE}_2(G)$ slightly exceeds $\frac{3}{2}|E(G)|$. These graphs can be constructed as follows. Let $x, y \geq 4$ be any two integers congruent to 0 modulo 4. The graph $H(x, y)$ is the graph obtained by starting from the disjoint union of a cycle C with length x and a cycle C' with length y, by adding an edge joining any vertex of C and any vertex of C'. Note that $H(x, y)$ has odd size.

Theorem 4.4.6. *Let x, y be any two integers congruent to 0 modulo 4, with $x, y \geq 4$. Then, we have* $\mathrm{mE}_2(H(x, y)) = \left\lceil \frac{3}{2}|E(H(x, y))| \right\rceil$.

Proof. We begin by showing the following claim:

Claim 4.4.7. *Let G be obtained from a cycle C with length x at least 4 congruent to 0 modulo 4 by adding an edge from any vertex v of C to a new pending vertex u. Then, by any proper 2-labelling ℓ of G, exactly half of the edges of G must be labelled 2. Furthermore, either:*

- *$\ell(vu) = 1$ and $c_\ell(v) = 5$, or*
- *$\ell(vu) = 2$ and $c_\ell(v)$ can be either of $4, 5, 6$.*

Proof of the claim. Let us denote by $v_0, \dots, v_{x-1}$ the successive vertices of C, where $v_0 = v$. Because $d(v_i) = 2$ for every $i \in \{1, \dots, x-1\}$, recall, according to Observation 2.1.5, that, by any proper 2-labelling ℓ of G, we must have $\ell(v_0v_1) \neq \ell(v_2v_3) \neq \ell(v_4v_5) \neq \dots \neq \ell(v_{x-2}v_{x-1})$ (and thus, by the length of x, we have $\ell(v_0v_1) \neq \ell(v_{x-2}v_{x-1})$), and similarly $\ell(v_1v_2) \neq \ell(v_3v_4) \neq \ell(v_5v_6) \neq \dots \neq \ell(v_{x-1}v_0)$ (and thus $\ell(v_1v_2) \neq \ell(v_{x-1}v_0)$). So there are essentially three ways for ℓ to be designed:

- If $\ell(v_0v_1) = \ell(v_0v_{x-1}) = 1$, then $\ell(v_1v_2) = \ell(v_{x-1}v_{x-2}) = 2$, and $c_\ell(v_1) = c_\ell(v_{x-1}) = 3$. In that case, so that $c_\ell(v_0) \neq 3$, we must have $\ell(v_0u) = 2$ in which case $c_\ell(v_0) = 4$.
- If $\ell(v_0v_1) = \ell(v_0v_{x-1}) = 2$, then $\ell(v_1v_2) = \ell(v_{x-1}v_{x-2}) = 1$, and $c_\ell(v_1) = c_\ell(v_{x-1}) = 3$. In that case, we can either have $\ell(v_0u) = 1$ in which case $c_\ell(v_0) = 5$, or $\ell(v_0u) = 2$ in which case $c_\ell(v_0) = 6$.

— If $\ell(v_0v_1) = 1$ and $\ell(v_0v_{x-1}) = 2$, then $\ell(v_1v_2) = 1$ and $\ell(v_{x-1}v_{x-2}) = 2$, and $c_\ell(v_1) = 2$ and $c_\ell(v_{x-1}) = 4$. In that case, so that $c_\ell(v_0) \neq 4$, we must have $\ell(v_0u) = 2$ in which case $c_\ell(v_0) = 5$.

This concludes the proof of the claim. $\diamond$

Let $G = H(x, y)$, and ℓ be a proper 2-labelling of G. Let H_1, H_2 be the two connected components resulting from the removal of the unique bridge uv of G, and G_1 and G_2 be the subgraphs $H_1 + uv$ and $H_2 + uv$, respectively, of G (where, say, G_1 contains the cycle C_1 with length x, and G_2 contains the cycle C_2 with length y). Applying Claim 4.4.7 onto G_1 and G_2 and the restriction of ℓ to these graphs, we deduce that we cannot have $\ell(uv) = 1$ as otherwise we would have $c_\ell(u) = c_\ell(v) = 5$, a contradiction. So we must have $\ell(uv) = 2$. Furthermore, still by Claim 4.4.7, exactly half of the edges of C_1 must be labelled 2 by ℓ, and similarly exactly half of the edges of C_2 must be labelled 2. It yields that $\sigma(\ell) = \lceil \frac{3}{2}|E(H(x, y))|\rceil$. Note that ℓ does exist, since G is not an odd multi-cactus (due to the presence of the bridge uv). In particular, the edges of C_1 and C_2 can be 2-labelled in such a way that $c_\ell(u)$ and $c_\ell(v)$ are two distinct values in $\{4, 5, 6\}$. $\square$

4.4.2.3 Improved upper bounds

As shown previously, it seems that, in general, for nice connected bipartite graphs the bound in Theorem 4.4.3 might not be optimal. Following our investigations in the previous section, we believe that perhaps studying Conjecture 2.1.6 could be the right direction to investigate. Recall that this conjecture states there is an absolute constant $c \geq 1$ such that, for every nice connected bipartite graph $G \in \mathcal{B}_2$, we have $\mathrm{mE}_2(G) \leq \frac{3}{2}|E(G)| + c$.

Towards Conjecture 2.1.6, in this section our aim is to improve Theorem 4.4.3 further for the bipartite graphs of $\mathcal{B}_2$. First off, we point out that the theoretical upper bound in Theorem 4.4.3 cannot be reached for a bipartite graph in $\mathcal{B}_2$.

Observation 4.4.8. *For every graph $G \in \mathcal{B}_2$, we have* $\mathrm{mE}_2(G) < 2|E(G)|$

Proof. By definition of $\mathcal{B}_1$, $\mathcal{B}_2$ and $\mathcal{B}_3$, since $G \notin \mathcal{B}_1$ the graph G is not locally irregular. Now, if $\mathrm{mE}_2(G) = 2|E(G)|$, then the only proper 2-labelling of G is the one assigning label 2 to all edges. For such a labelling to be proper, G must have no two adjacent vertices having the same degree. So G must be locally irregular, a contradiction. $\square$

In particular contexts, better bounds can be obtained by adapting the arguments from the proof of Theorem 4.4.1 in a particular way.

Theorem 4.4.9. *Let $G = (U, V, E)$ be a nice connected bipartite graph where $|U|$ is even. Then, we have* $\mathrm{mE}_2(G) \leq |E(G)| + |V(G)| - 1$.

Proof. Let us denote by U_e (U_o, respectively) the set of vertices of U having even (odd, respectively) degree in G, and similarly by V_e (V_o, respectively) the set of vertices of V having even (odd, respectively) degree in G. Note that either $|U_e|$ and $|V_o|$ must have the same parity, or $|U_o|$ and $|V_e|$ must have the same parity. This is because, otherwise, since $|U|$ is even and $|U| = |U_e| + |U_o|$, the sizes $|U_e|$ and $|U_o|$ must have the same parity, we would get that also $|V_e|$ and $|V_o|$ have the same parity. From this, we would deduce that $\sum_{u \in U} d(u) \not\equiv \sum_{v \in V} d(v) \bmod 2$, which is not possible.

Without loss of generality, we may assume that U_e and V_o have the same parity, thus that $|U_e| + |V_o|$ is even. Our aim now, is to design a 2-labelling of G where all vertices in U get odd colour while all vertices in V get even colour. Such a labelling will obviously be proper. To that aim, we proceed as follows. Let us start with assigning label 1 to all edges of G. This way, at this point the colour of every vertex is exactly its degree; so all vertices in U_o and V_e verify the desired colour property, while all vertices in U_e and V_o do not. To fix these vertices, we consider any spanning tree T of G. We now repeatedly apply the following fixing procedure: we consider any two vertices x and y of $U_e \cup V_o$ that remain to be fixed, and flip (*i.e.*, turn the 1s into 2s, and *vice versa*) the labels of all edges on the unique path in T from x to y. This way, note that only the colours of x and y are altered modulo 2. Since $|U_e| + |V_o|$ is even, there are an even number of vertices to fix, and, by flipping labels along paths of T, we can fix the colour of all vertices in $U_e \cup V_o$. This results in a 2-labelling ℓ of G, with the desired properties, which is thus proper.

Note now that ℓ assigns label 2 only to a subset of the edges of T. Since T has $|V(G)| - 1$ edges, the result follows. □

Note for instance that, for a graph G, we have $|E(G)| + |V(G)| - 1 \leq \frac{3}{2}|E(G)|$ as soon as $|E(G)| \geq 2|V(G)| - 2$. As notable consequences, this implies that a connected bipartite graph $G \in \mathcal{B}_2$ with a part of even size verifies $\mathrm{mE}_2(G) \leq \frac{3}{2}|E(G)|$ as soon as G has minimum degree at least 4, or more generally when the graph is dense enough.

The same result also holds when G is bipartite and cubic (in which case $\chi_\Sigma(G) = 2$, by definition of odd multi-cacti), from a more general argument:

Observation 4.4.10. *Let G be a connected regular graph with $\chi_\Sigma(G) = 2$. Then, we have* $\mathrm{mE}(G) \leq \mathrm{mE}_2(G) \leq \frac{3}{2}|E(G)|$.

Proof. Let ℓ be a proper 2-labelling of G. Since G is regular, the edges labelled 1 by ℓ, and similarly the edges labelled 2, must induce a locally irregular subgraph of G. Then the 2-labelling ℓ' of G obtained by turning all 1s into 2s, and *vice versa*, is also proper. Now there is one of ℓ and ℓ' that assigns label 2 to at most half of the edges, and the conclusion follows. □

Slight modifications of the proof of Theorem 4.4.9 also yield the desired result for certain bipartite graphs that are Hamiltonian.

Observation 4.4.11. *Let $G = (U, V, E)$ be a Hamiltonian bipartite graph where $|U|$ is even. Then, we have* $\mathrm{mE}(G) \leq \mathrm{mE}_2(G) \leq \frac{3}{2}|E|$.

Proof. Just mimic the proof of Theorem 4.4.9, but repair pairs of defective vertices of G along a Hamiltonian cycle $C = (v_0, \dots, v_{n-1}, v_0)$, matching each of them, say, with the next defective vertex in the ordering of C. If this fixing process turns more than half of the edges to 2, then, instead, repair pairs of vertices around C matching each of them with the previous defective vertex in the ordering (which is equivalent to flipping the labels along C). □

4.4.2.4 Trees

Our main result here is that for every nice tree T, we have $\mathrm{mE}_2(T) \leq \frac{3}{2}|E(G)|$, which cannot be lowered in general, due to Corollary 4.4.5. Still, it confirms Conjecture 2.1.6 for nice trees. Let us recall that it was proved in [18] that nice forests admit equitable proper 2-labelling. This directly implies our result below for trees with even size, while it does not for trees with odd size (as a 2-labelling where the number of assigned 2s is one more than the number of assigned 1s does not fulfil our claim).

Theorem 4.4.12. *For every nice tree T, we have* $\mathrm{mE}_2(T) \leq \frac{3}{2}|E(G)|$.

Proof. The proof is by induction on the number of branching vertices of T, where by branching vertex we mean any vertex of degree at least 3. The base case is when T has no branching vertex, *i.e.*, when $\Delta(T) \leq 2$. In that case, T is a path. Let us here consider the 2-labelling ℓ of T obtained by assigning labels $1, 1, 2, 2, 1, 1, 2, 2, \dots$ as traversing the edges from an end-vertex to the second one. It follows from Observations 2.1.5 and 2.2.4 that ℓ is proper. Lastly, since 1s and 2s are assigned by pairs starting from a pair of 1s, it should be clear that ℓ assigns more 1s than 2s. Thus, $\mathrm{mE}_2(T) \leq \sigma(\ell) \leq \frac{3}{2}|E|$.

We now focus on the general case. That is, we now assume that T has branching vertices, and every nice tree with fewer branching vertices verifies the claim. Let us root T at some degree-1 vertex r. In the usual way, this defines a (virtual) orientation of T. By a *deepest branching vertex* of T, we refer to a branching vertex whose all descendants are not branching vertices, *i.e.*, they have degree at most 2.

Let us consider a deepest branching vertex v of T. Then v is adjacent to its parent w and there are $k \geq 2$ hanging paths $P_1, \dots, P_k$ attached to v. Note that some of the P_i's may be of length 1 in case some of the children of T are leaves. Let T' be the tree obtained from T by removing the edges of $P_1, \dots, P_k$ (*i.e.*, all their vertices different from v). If T' is just an edge, then T is actually a subdivided star. If T is a star with at least two leaves, then it is locally irregular and we can assign label 1 to all edges. Otherwise, when T is a subdivided star different from a star, then, without loss of generality, P_1 has length at least 2. We then change the root r to be the degree-1 vertex of P_1 so that, now, we can assume that T' indeed is not an edge.

Thus, we can assume that T' is not just an edge. Since T' has less branching vertices than T, by the induction hypothesis there is a proper 2-labelling ℓ' of T' verifying $\mathrm{mE}(T', \ell') \leq \frac{3}{2}|E(T')|$. We wish to extend ℓ' to the edges of $P_1, \dots, P_k$, thus to a 2-labelling ℓ of T. To that aim, we consider the following two extension schemes for extending ℓ' to the edges of one P_x of the P_i's:

— 1-*extension*: We assign labels $1, 1, 2, 2, 1, 1, 2, 2, \dots$ to the consecutive edges of P_x, as they are traversed going from v to the degree-1 vertex of P_x.

— 2-*extension*: We assign labels $2, 1, 1, 2, 2, 1, 1, 2, 2, \dots$ to the consecutive edges of P_x, as they are traversed going from v to the degree-1 vertex of P_x.

Note that whenever P_x has length not congruent to 1 modulo 4, the number of assigned 1s is always at least the number of assigned 2s by both 1-extensions and 2-extensions. More precisely, if P_x has length congruent to 1 modulo 4, then the number of 2s by a 2-extension is one more than the number of 1s, and *vice versa* by a 1-extension. Recall also that 1-vertices cannot be involved in colour conflicts. Furthermore, for two adjacent 2-vertices x, y to have the same colour, the edge incident to x different from xy must be labelled the same way as the edge incident to y different from xy. From this, we deduce that when extending ℓ' to the edges of the P_i's via 1-extensions and 2-extensions, we must just make sure that 1) the colour of v does not get equal to the colour of its parent w, and 2) the colour of v does not get equal to the colour of one of its children.

We note that the second type of colour conflict cannot actually occur. Indeed, note that by a 1-extension of P_x, the neighbour of v in P_x, unless it has degree 1 (in which case it cannot be in conflict with v), gets colour 2, while, by a 2-extension, it gets colour 3. Since v is a branching vertex with $k \geq 2$ children, thus of degree $k + 1$, when performing 1-extensions and 2-extensions to the P_i's, vertex v gets colour precisely $k+1 \geq 3$ if only 1-extensions are performed, and colour at least $k + 2 \geq 4$ if at least one 2-extension is performed.

Thus, we just need to find a combination of 1-extensions and 2-extensions to the P_i's so that no colour conflict involving v and its parent w arises. Also, we need to make sure that the number of assigned 1s is at least the number of assigned 2s. If one of the P_i's has length not congruent to 1 modulo 4, then we choose it as P_1. Otherwise, if they all have length congruent to 1 modulo 4, then we choose any P_i as P_1.

We first perform 1-extensions only, *i.e.*, to all P_i's. If the colour of v gets different from that of w, then we are done. Otherwise, when performing a 2-extension to P_1 and a 1-extension to all other P_i's, the colour of v gets bigger, thus getting different from the colour of w. This results in the desired extension ℓ to all edges of T.

Let us conclude by noting that the number of 1s assigned by ℓ is at least the number of assigned 2s. This is because $\mathrm{mE}_2(T', \ell') \leq \frac{3}{2}|E(T')|$, and, as mentioned earlier, by 1-extensions to the P_i's the number of assigned 1s is at least the number of assigned 2s. By 2-extensions, this is true when performed on paths of length not congruent to 1 modulo 4. By our choice of P_1, if P_1 has length congruent to 1 modulo 4, then so do all P_i's. In that precise case, the number of 2s assigned to the edges of P_1 is one bigger than the number of assigned 1s, but this is compensated by the fact that, in P_2, the number of assigned 1s is one bigger than the number of assigned 2s. Thus we additionally have $\sigma(\ell) \leq \frac{3}{2}|E(T)|$, as desired. □

4.5 Conclusion

In this chapter, we have studied proper labellings of graphs with the additional requirement that we want the sum of assigned labels to be as small as possible. Our interests were guided by both straight questions, such as determining $\mathrm{mE}(G)$ for a given graph G, as well as more fundamental ones, such as the difference, in general, between $\mathrm{mE}_k(G)$ and $\mathrm{mE}_{k'}(G)$ for $k \neq k'$. We have also investigated the complexity of finding proper labellings that also minimise the sum of labels being used.

We quickly ran into Conjecture 2.1.4, which seems rather natural, considering how plausible the 1-2-3 Conjecture seems to be, and that graphs, in general, seem to need only a few 3s to design proper 3-labellings. Conjecture 2.1.4 stands as the main open problem regarding our investigations in the current chapter. It would also be interesting to progress towards its refinement for bipartite graphs, Conjecture 2.1.6. A way to progress towards answering both questions could be to exhibit families of connected (possibly bipartite) graphs G for which $\mathrm{mE}(G)$ is "large", *i.e.*, larger than the quantity in Theorem 4.4.6.

Regarding our algorithmic results in Section 4.2, we note that they all deal, for a given graph G, with the parameter $\mathrm{mE}_k(G)$ (for some k), and not with the more general parameter $\mathrm{mE}(G)$. This is mainly because, as indicated by Theorem 4.3.5, in general there is no absolute constant that bounds, for all graphs G, the smallest k such that $\mathrm{mE}(G) = \mathrm{mE}_k(G)$. In particular, even for a graph G of bounded treewidth, although we can determine $\mathrm{mE}_k(G)$ in polynomial time for any fixed k (due to our algorithm in Theorem 4.2.5), running multiple iterations of our algorithm to determine $\mathrm{mE}(G)$ is not feasible in polynomial time. Thus, we leave the following problem open even for the seemingly simplest case:

Question 4.5.1. *What is the complexity of determining* $\mathrm{mE}(T)$ *for a given tree* T*?*

CHAPTER 5

Minimising the number of edges labelled 3

An intuition from previous investigations on the 1-2-3 Conjecture is that, in general, it should always be possible to produce proper 3-labellings assigning label 3 to only a few edges. In this chapter we investigate proper 3-labellings of graphs that also minimise the number of edges labelled 3.
We prove that, for every $p \geq 0$, there are various graphs needing at least p 3s in their proper 3-labellings. Actually, deciding whether a given graph can be properly 3-labelled with p 3s is NP-complete for every $p \geq 0$. We also focus on classes of 3-chromatic graphs. For various classes of such graphs (cacti, cubic graphs, triangle-free planar graphs, etc.), we prove that there is no $p \geq 1$ such that all their graphs admit proper 3-labellings assigning label 3 to at most p edges. In such cases, we provide lower and upper bounds on the number of 3s needed.
This chapter presents a joint work with J. Bensmail and F. Mc Inerney, published in [22] and presented in [23].

Our goal in this chapter is to study and formally establish the intuition that, in general, graphs should admit proper 3-labellings assigning only a few 3s. We study this through two questions.

— The very first question to consider is whether, given a (possibly infinite) class $\mathcal{F}$ of graphs, the members of $\mathcal{F}$ admit proper 3-labellings assigning only a constant number of 3s, *i.e.*, whether there is a constant $c_{\mathcal{F}} \geq 0$ such that all graphs of $\mathcal{F}$ admit proper 3-labellings assigning label 3 to at most $c_{\mathcal{F}}$ edges. Note that this is something that is already known to hold for a few graph classes. For instance, all nice trees admit proper 2-labellings (recall Proposition 2.2.3), thus proper 3-labellings assigning label 3 to no edge. Similarly, from Theorem 4.4.3, it can be deduced that all nice bipartite graphs admit proper 3-labellings assigning label 3 to at most two edges.

— In case $\mathcal{F}$ admits no such constant $c_{\mathcal{F}}$, *i.e.*, the number of 3s the members of $\mathcal{F}$ need in their proper 3-labellings is a function of their number of edges, the second question we consider is whether the number of 3s needed must be "large" for a given member of $\mathcal{F}$, with respect to the number of its edges.

Throughout this chapter, we investigate these two questions in general and for more restricted classes of graphs. We start off in Section 5.1 by raising preliminary observations and results. Then, in Section 5.2, we introduce proof techniques for establishing lower and upper bounds on the number of 3s needed to construct proper 3-labellings for some graph classes. In Sections 5.3 and 5.4, we use these tools to establish that, for several classes of graphs, the number of 3s needed in their proper 3-labellings is not bounded by an absolute constant. In such cases, we exhibit bounds (functions depending on the size of the considered graphs) on this number. The difference between these two sections is that in Section 5.4 we only provide either upper or lower bounds on the studied parameter for the families of graphs under consideration, while in Section 5.3 we provide both upper and lower bounds. Our results lead us to propose Conjecture 2.1.9, claiming that every nice graph G admits proper 3-labellings that assign label 3 to at most one third the edges of G.

5.1 Preliminary results

Let us briefly recall the definitions given in Section 2.1. We denote by $\mathrm{mT}(G)$ the minimum number of edges assigned label 3 by a proper 3-labelling of a graph G. Also, for a class $\mathcal{F}$ of graphs, $\mathrm{mT}(\mathcal{F})$ is defined as the maximum value of $\mathrm{mT}(G)$ over the members G of $\mathcal{F}$. Also, for every $p \geq 0$, we denote by $\mathcal{G}_p$ the class of graphs G with $\mathrm{mT}(G) = p$. For convenience, we also define $\mathcal{G}_{\leq p} = \mathcal{G}_0 \cup \cdots \cup \mathcal{G}_p$.

Let us now employ the above notations to restate Theorem 4.4.3:

Theorem 5.1.1. *If G is a nice bipartite graph, then $G \in \mathcal{G}_{\leq 2}$. More precisely, $G \in \mathcal{G}_0$ if G is not an odd multi-cactus, $G \in \mathcal{G}_2$ if G is a cycle of length congruent to 2 modulo 4, and $G \in \mathcal{G}_1$ otherwise (i.e., if G is an odd multi-cactus different from a cycle C_{4k+2}).*

Theorem 5.1.1 is worrisome in the sense that, even without considering any additional constraint, we do not know much about how proper 3-labellings behave beyond the scope of bipartite graphs. Our take in this chapter is to focus on the next natural case to consider, that of 3-chromatic graphs, which fulfil the 1-2-3 Conjecture (recall Theorem 2.2.1). Unfortunately, as will be seen later on, a result equivalent to Theorem 5.1.1 for 3-chromatic graphs does not exist, even for very restricted classes of 3-chromatic graphs (*e.g.*, cacti, cubic graphs, triangle-free planar graphs, etc.).

Regarding the classes $\mathcal{G}_0, \mathcal{G}_1, \ldots$, it is worth mentioning right away that each $\mathcal{G}_p$ is well-populated, in the sense that there exist infinitely many graphs, with various properties, belonging to $\mathcal{G}_p$. Actually, it turns out that deciding whether a given graph G belongs to $\mathcal{G}_{\leq p}$ is NP-complete for every $p \geq 0$. We postpone the proofs of these statements to Section 5.2 (Theorems 5.2.3 and 5.2.4), as they require the tools and results introduced earlier in the same section.

As mentioned earlier, we will see throughout this chapter that, for several graph classes $\mathcal{F}$, there is no $p \geq 0$ such that $\mathcal{F} \subset \mathcal{G}_{\leq p}$. For such a class, we want to know whether the proper 3-labellings of their members require assigning label 3 many times, with respect to their number of edges. We study this aspect through the following terminology, already introduced in Chapter 2, Definition 2.1.8. For a nice graph G, we define $\rho_3(G) = \mathrm{mT}(G)/|E(G)|$. We extend this ratio to a class $\mathcal{F}$ by setting $\rho_3(\mathcal{F}) = \sup\{\rho_3(G) : G \in \mathcal{F}\}$. This parameter is at the heart of Conjecture 2.1.9 which we introduced in Section 2.1, claiming that every nice graph G verifies $\rho_3(G) \leq \frac{1}{3}$.

Recall that Conjecture 2.1.9 can be seen as a weaker version of Conjecture 2.1.2, investigated in [18] and in Section 3. However, we cannot benefit much from the results in [18] or Section 3, since most of these results are about equitable proper 3-labellings of classes of bipartite graphs, while bipartite graphs form a pretty well-understood case in the context of this chapter (recall Theorem 5.1.1).

One result we do get from [18] is an upper bound on ρ_3 for complete graphs, which is actually improved by another result presented in Chapter 4. Indeed, it was shown in [18] that complete graphs K_n with $n \geq 5$ admit equitable proper 3-labellings, which implies that they verify Conjecture 2.1.9, *i.e.*, $\mathrm{mT}(K_n) \leq |E(K_n)|/3$ which is roughly of order $n^2/6$. Recall now Theorem 4.1.5, where we exhibited proper 3-labellings of complete graphs where the sum of assigned labels is as small as possible. Looking closely at the proof, it turns out that the designed proper 3-labellings assign label 3 to roughly $n/4$ edges, which yields a better upper bound on $\rho_3(K_n)$. Determining the precise ratio in general sounds like an interesting challenge. Through computer experimentation, we were able to verify that $K_n \in \mathcal{G}_1$ for $3 \leq n \leq 5$, while $K_n \in \mathcal{G}_2$ for $6 \leq n \leq 9$, and $K_n \in \mathcal{G}_3$ for $10 \leq n \leq 12$. However, we did not manage to prove a general result. We are not even sure if there exists a $p \geq 3$ such that all complete graphs are in $\mathcal{G}_{\leq p}$.

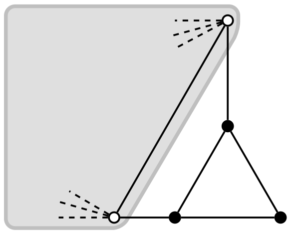
We will now present some initial results on proper labellings, which will be useful in the next sections.

Let ℓ be a k-labelling of some graph, and let $\sigma : \{1, \dots, k\} \to \{1, \dots, k\}$ be a permutation of $\{1, \dots, k\}$. We denote by $\mathrm{sw}(\ell, \sigma)$ the k-labelling obtained from ℓ by switching labels as indicated by σ. That is, if $\ell(e) = i$ for some edge e and label i, then $\mathrm{sw}(\ell, \sigma)(e) = \sigma(i)$. Assuming the set of labels $\{1, \dots, k\}$ is clear from the context, for any two $i, j \in \{1, \dots, k\}$, we denote by $\sigma_{i\leftrightarrow j}$ the permutation only swapping labels i and j. That is, $\sigma_{i\leftrightarrow j}(i) = j$, $\sigma_{i\leftrightarrow j}(j) = i$, and $\sigma_{i\leftrightarrow j}(l) = l$ for every $l \in \{1, \dots, k\} \setminus \{i, j\}$.

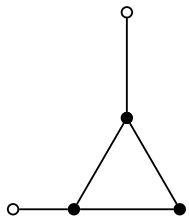
Recall the definition of a d-regular graph G. We extend this definition as follows: we say that a graph G is *quasi d-regular* if every vertex $v \in V(G)$ satisfies $d_G(v) \in \{1, d\}$. Clearly, every graph that is d-regular, is also quasi d-regular.

Lemma 5.1.2. *If ℓ is a proper 3-labelling of a quasi d-regular graph G, then $\mathrm{sw}(\ell, \sigma_{1\leftrightarrow 3})$ is also proper.*

Proof. Assume G is quasi d-regular for some $d \geq 2$, and set $\ell' = \mathrm{sw}(\ell, \sigma_{1\leftrightarrow 3})$. Recall that, by a k-labelling, a vertex of degree 1 can never be involved in a colour conflict with its neighbour. Consider any edge $vw \in E(G)$ with $d_G(v) = d_G(w) = d$. For $1 \leq i \leq 3$, let n_i be the number of edges incident to v that are labelled i by ℓ. Then, $n_1 + n_2 + n_3 = d$, $c_\ell(v) = n_1 + 2n_2 + 3n_3$, and



(a) G



(b) H

Figure 5.1 – A graph G containing another graph H as a weakly induced subgraph. In G, the white vertices can have arbitrarily many neighbours in the grey part, while the full neighbourhoods of the black vertices are as displayed. In H, the white vertices are the border vertices, while the black vertices are the core vertices.

$c_{\ell'}(v) = 3n_1 + 2n_2 + n_3$, and thus, $c_\ell(v) + c_{\ell'}(v) = 4(n_1 + n_2 + n_3) = 4d$. Similarly, we have that $c_\ell(w) + c_{\ell'}(w) = 4d$. Therefore, $c_\ell(v) - c_\ell(w) = c_{\ell'}(w) - c_{\ell'}(v)$, with $c_\ell(v) \neq c_\ell(w)$ (since ℓ is a proper labelling) implying that $c_{\ell'}(w) \neq c_{\ell'}(v)$. It follows that ℓ' is a proper 3-labelling of G. □

Analogously, one can prove:

Lemma 5.1.3. *If ℓ is a proper 2-labelling of a quasi 3-regular graph, then $\mathrm{sw}(\ell, \sigma_{1\leftrightarrow 2})$ is also proper.*

5.2 Tools for establishing bounds on mT and ρ_3

5.2.1 Weakly induced subgraphs – A tool for lower bounds

Most of the lower bounds on mT and ρ_3 that we exhibit in Section 5.3 are through a particular graph construction. The general idea is that, if we have a collection of graphs $H_1, \dots, H_n$ with certain structural and labelling properties, then it is possible to combine these H_i's in some fashion to form a bigger graph G in which the H_i's retain their respective labelling properties, from which we can deduce that G itself has certain labelling properties.

In order to state this construction formally, we need to introduce some terminology first (see Figure 5.1 for an illustration).

Definition 5.2.1. *Let G and H be two graphs such that $V(H) \subseteq V(G)$. We say that G contains H as a weakly induced subgraph if, for every vertex $v \in V(H)$, either $d_H(v) = 1$ or $d_H(v) = d_G(v)$. For every edge $uv \in E(G)$, if $u \in V(H)$ and $v \in V(G) \setminus V(H)$, then $d_H(u) = 1$; we call these the* border vertices *of H. Also, we call the other vertices of H (i.e., those that are not border vertices) its* core vertices.

By definition, note that if G contains H as a weakly induced subgraph and $\delta(H) \geq 2$, then H is a collection of connected components of G. In particular, if G is a connected graph, then H is isomorphic to G. For this reason, this notion makes more sense when $\delta(H) = 1$.

Let H_1, H_2 be two weakly induced subgraphs of a graph G. We say that H_1 and H_2 are *disjoint* (in G) if they share no core vertices. It follows directly from the definition that, for every $v \in V(G)$, if $v \in V(H_1) \cap V(H_2)$, then v is a border vertex of both H_1 and H_2. For a labelling

ℓ of G and a subgraph H of G, we denote by $\ell|_H$ the labelling of H inferred from ℓ, *i.e.*, we have $\ell|_H(e) = \ell(e)$ for every edge $e \in E(H)$.

The key result is that, if a graph G contains other graphs $H_1, \dots, H_n$ as pairwise disjoint weakly induced subgraphs, then the labelling properties of the H_i's, in particular $\mathrm{mT}(H_i)$, can be inferred to those of G:

Lemma 5.2.2. *Let G be a graph containing nice graphs $H_1, \dots, H_n$ as pairwise disjoint weakly induced subgraphs. If ℓ is a proper 3-labelling of G, then $\ell|_{H_i}$ is a proper 3-labelling of H_i for every $i \in \{1, \dots, n\}$. Consequently,* $\mathrm{mT}(G) \geq \sum_{i=1}^{n} \mathrm{mT}(H_i)$.

Proof. Consider H_j for some $1 \leq j \leq n$. Since, by any k-labelling of a nice graph, a vertex of degree 1 cannot get the same colour as its unique neighbour, then it cannot be involved in a colouring conflict. This implies that $\ell|_{H_j}$ is proper if and only if any two adjacent core vertices of H_j get distinct colours by $\ell|_{H_j}$. By the definition of a weakly induced subgraph, we have $d_{H_j}(v) = d_G(v)$ for every core vertex v of H_j, which implies that $c_{\ell|_{H_j}}(v) = c_\ell(v)$. Thus, for every edge $uv \in E(H_j)$ joining core vertices, we have $c_\ell(u) = c_{\ell|_{H_j}}(u) \neq c_{\ell|_{H_j}}(v) = c_\ell(v)$ since ℓ is proper, meaning that $\ell|_{H_j}$ is also proper. Now, since G contains nice graphs $H_1, \dots, H_n$ as pairwise disjoint weakly induced subgraphs, then $\mathrm{mT}(G) \geq \sum_{i=1}^{n} \mathrm{mT}(H_i)$. □

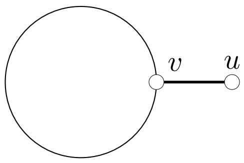
Through an easy use of Lemma 5.2.2, we can already establish results of interest regarding the parameter mT. For instance, we can prove that each graph class $\mathcal{G}_p$ ($p \geq 1$) contains infinitely many graphs with various properties.

Theorem 5.2.3. *$\mathcal{G}_p$ contains infinitely many graphs for every $p \geq 0$.*

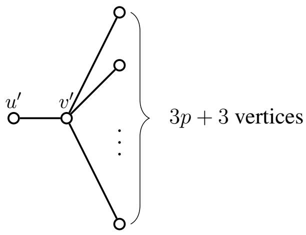
Proof. Clearly, the statement is true for $p = 0$. We now show that it is also true for every $p \geq 1$. Let H be a graph with $\delta(H) = 1$ and $\mathrm{mT}(H) = 1$ (such graphs exist, see, *e.g.*, our results from Section 5.3, in particular the graphs illustrated in Figure 5.3). Let uv be an edge of H such that $d_H(u) = 1$ and $d_H(v) \geq 2$. Also, let T be any locally irregular graph with an edge $u'v'$ such that $d_T(u') = 1$ and $d_T(v') \geq 3p + 3$.

Now, let G be the graph that is the disjoint union of T and of p copies $H_1, \dots, H_p$ of H, and identify u' and the p copies of u to a single vertex w (see Figure 5.2 for an illustration of G). Clearly, G contains T and the disjoint union of p copies of H as pairwise disjoint weakly induced subgraphs (with a slight abuse of notations, for simplicity we refer to both the original T and its copy in G as T). By Lemma 5.2.2, we have $\mathrm{mT}(G) \geq \mathrm{mT}(T) + p \cdot \mathrm{mT}(H) = p$ since T is locally irregular (thus, $\mathrm{mT}(T) = 0$) and $\mathrm{mT}(H) = 1$.

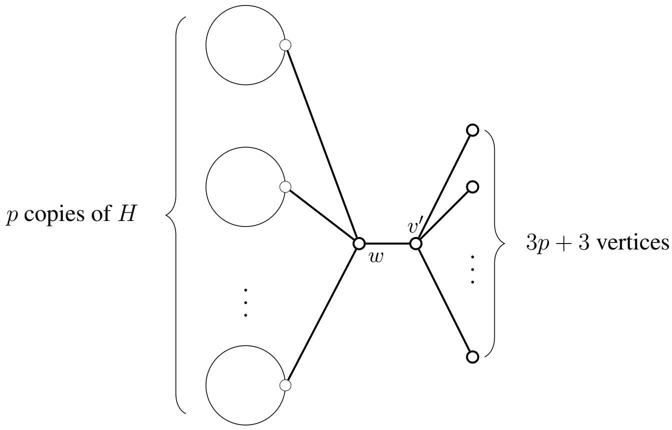
To prove that the equality actually holds, it suffices to construct a proper 3-labelling ℓ of G with $\mathrm{nb}_\ell(3) = p$. Recall that $\mathrm{nb}_\ell(\alpha)$ is used to denote the number of times that ℓ assigns label α on the edges of G. Let ℓ' be a proper 3-labelling of H such that $\mathrm{nb}_{\ell'}(3) = 1$, which exists since $\mathrm{mT}(H) = 1$. To obtain ℓ, for each H_i, we set $\ell(e) = \ell'(e)$ for every edge e of H_i, while we set $\ell(e) = j$ for every edge e of T, where $j \in \{1, 2\}$ is chosen so that $c_\ell(w) \neq c_{\ell'}(v)$ for v in each copy of H_i (recall that $c_{\ell'}(v)$ is the same for each copy of H_i). As a result, for any H_i, for every vertex $x \neq w$ of H_i, we get $c_\ell(x) = c_{\ell'}(x)$. Hence, for any H_i, for every edge xy of H_i not containing w, we have $c_\ell(x) \neq c_\ell(y)$. Furthermore, for every vertex x of T different from w, we have either $c_\ell(x) = d_G(x)$ or $c_\ell(x) = 2d_G(x)$, meaning that, for every edge xy of T not containing w, we have $c_\ell(x) \neq c_\ell(y)$ since T is locally irregular. Now, by the construction of ℓ, note that w cannot be in conflict with its neighbours in the H_i's (due to the choice of j), and $c_\ell(w) < 3p + 3 \leq d_G(v') \leq c_\ell(v')$, meaning that w and v' cannot be in conflict. Thus, ℓ is proper.



(a) A graph H s.t. $\mathrm{mT}(H) \geq 1$ and $\delta(H) = 1$



(b) A locally irregular graph T



(c) The constructed graph G

Figure 5.2 – An illustration of the construction described in the proof of Theorem 5.2.3. The graph H is abstractedly represented by a cycle, with the edge uv being such that $d_H(u) = 1$ and $d_H(v) \geq 2$. The graph T is chosen to be the star on $3p + 5$ vertices, for any $p \geq 1$.

□

Note that, in the proof above, the structure of T does not matter, and can be anything as long as T is locally irregular and has the particular edge $u'v'$ with $d_T(u') = 1$ and $d_T(v') \geq 3p + 3$. In particular, T can potentially contain any graph as an induced subgraph. Thus, each graph class $\mathcal{G}_p$ ($p \geq 1$) contains infinitely many graphs with various properties.

Using similar ideas, we can actually prove that deciding if a graph G belongs to $\mathcal{G}_p$ cannot be done in polynomial time, unless P=NP.

Theorem 5.2.4. *Given a graph G and any (fixed) integer $p \geq 1$, deciding if $G \in \mathcal{G}_{\leq p}$ is NP-complete.*

Proof. The problem is obviously in NP. Let us focus on proving it is also NP-hard. This is done by a reduction from the 2-LABELLING problem, which was proved to be NP-hard, *e.g.*, in [64]. In that problem, a graph H is given, and the goal is to decide whether H admits proper 2-labellings. Given an instance H of 2-LABELLING, we construct, in polynomial time, a graph G such that $\mathrm{mT}(G) = p$ if and only if H admits proper 2-labellings.

Looking closely at the proof from [64], it can be noted that 2-LABELLING remains NP-hard when restricted to graphs with minimum degree 1. Thus, we can assume H has this property.

The construction of G is achieved as follows. Let H' be a graph with $\delta(H') = 1$ and $\mathrm{mT}(H') = 1$ (as mentioned in the proof of Theorem 5.2.3, such graphs exist, and two are illustrated in Figure 5.3). Let uv be an edge of H' such that $d_{H'}(u) = 1$ and $d_{H'}(v) \geq 2$. Now, start from G being the disjoint union of H and of p copies $H'_1, \dots, H'_p$ of H', and then identify a vertex of degree 1 of H and of the p copies of u to a single vertex w. Finally, attach new vertices of degree 1 to w so that the degree of w in G gets at least four times bigger than the degree of any of its neighbours. Clearly, the construction of G is achieved in polynomial time.

We now prove the equivalence between the two problems.

- Assume ℓ is a proper 3-labelling of G such that $\mathrm{nb}_\ell(3) = p$. Note that G contains H and p copies of H' as pairwise disjoint weakly induced subgraphs. Due to Lemma 5.2.2, and because $\mathrm{mT}(H') = 1$, this means that we must have $\mathrm{nb}_{\ell|_{H'_i}}(3) = 1$ for every $i \in \{1, \dots, p\}$, and, thus, $\mathrm{nb}_{\ell|_H}(3) = 0$. Then, $\ell|_H$ must be a proper 2-labelling of H.
- Assume ℓ is a proper 2-labelling of H. Since $\mathrm{mT}(H') = 1$, there exists a proper 3-labelling ℓ' of H' where $\mathrm{nb}_{\ell'}(3) = 1$. Now, let ℓ'' be the 3-labelling of G obtained by setting $\ell''(e) = \ell(e)$ for every $e \in E(H)$, setting $\ell''(e) = \ell'(e)$ for every $e \in E(H'_i)$ for each $i \in \{1, \dots, p\}$, and setting $\ell''(e) = 1$ for every remaining pending edge attached at w. By the properties of ℓ and ℓ', and by arguments similar to those used in the proof of Theorem 5.2.3, no conflict can occur along an edge not containing w. Now, regarding w, due to its degree, it follows that $c_{\ell''}(w)$ must be strictly bigger than the colour of each of the neighbours of w. Thus, ℓ'' is a proper 3-labelling of G, and $\mathrm{nb}_{\ell''}(3) = p$.

□

To close this section, we point out that, in some contexts, we can add some structure to a given graph without altering its value of mT. In some of the later proofs, this will be particularly convenient for applying inductive arguments or simplifying the structure of a considered graph.

Lemma 5.2.5. *Let G be a nice graph with minimum degree 1 and $v \in V(G)$ be such that $d_G(v) = 1$. If G' is the graph obtained from G by adding $x > 0$ vertices of degree 1 adjacent to v, then $\mathrm{mT}(G') = \mathrm{mT}(G)$.*

Proof. Since G' contains G as a weakly induced subgraph, then by Lemma 5.2.2, we have that $\mathrm{mT}(G') \geq \mathrm{mT}(G)$. To show that $\mathrm{mT}(G') \leq \mathrm{mT}(G)$, it suffices to extend a proper 3-labelling of G to one of G' that assigns label 3 to the same number of edges. To do this, simply note that since each one of the leaves adjacent to v has degree 1, its colour cannot be in conflict with that of v. Thus, the only colour conflict that can occur when extending the labelling, is between v and its unique neighbour in G. If, by labelling all of the edges incident to the leaves adjacent to v with 1s, there is a colour conflict between v and its neighbour in G, then it suffices to change exactly one of those labels to 2. □

5.2.2 Partitioning into stable sets – A tool for upper bounds

Due to Theorem 5.1.1, investigating the parameters mT and ρ_3 is interesting for graphs with chromatic number at least 3, *i.e.*, that are not bipartite. These graphs have odd-length cycles. We take advantage of these cycles, in the sense explained in Section 2.2.1, to prove the following upper bound on ρ_3 for 3-chromatic graphs.

Theorem 5.2.6. *If G is a connected 3-chromatic graph, then $\rho_3(G) \leq |V(G)|/|E(G)|$.*

Proof. The first part of this proof consists in creating a proper 3-labelling of G, exactly as it is described in the proof of Theorem 2.2.1. Let ℓ be the resulting labelling.

Recall that we have $\ell(e) = 2$ for every $e \in E(G) \setminus E(H)$, where H is the odd unicyclic spanning subgraph of G constructed in the first part of the proof of Theorem 2.2.1. Thus, only the edges of H can be assigned label 0 by ℓ. Since $|E(H)| = |V(G)|$ and we can replace all assigned 0s with 3s without breaking the modulo 3 property, we have $\mathrm{mT}(G) \leq |V(G)|$, which implies that $\rho_3(G) \leq |V(G)|/|E(G)|$. □

Theorem 5.2.6, by itself, has implications on Conjecture 2.1.9. In particular, every sufficiently dense connected 3-chromatic graph verifies the conjecture. This remark applies to, *e.g.*, every connected 3-chromatic graph G with $\delta(G) \geq 6$, since it obviously verifies $|E(G)| \geq 3|V(G)|$. Note that, in that case, the connectivity condition can actually be dropped, as every connected component of a 3-chromatic graph is 3-colourable (so, for each component, one of Theorems 5.1.1 and 5.2.6 applies).

Corollary 5.2.7. *If G is a 3-chromatic graph with $\delta(G) \geq 6$, then $\rho_3(G) \leq 1/3$.*

In general, and more particularly for less dense graphs, it would be interesting to find ways to improve the arguments in the proof of Theorem 5.2.6 to further reduce the number of assigned 3s. Note that several of our arguments could actually be subject to improvement. For instance, in the current proof, we always set $\ell(e) = 2$ for an edge $e \in E(G) \setminus E(H)$, which might be one of the reasons why many 3s might appear through the eventual walk-switching procedure. It seems, however, that in general, this is tough to improve upon significantly without further assumptions on G. Similarly, in some contexts, it might be possible to choose the odd unicyclic subgraph H of G in a clever way, but this seems hard to do in general. A more interesting direction is about choosing the proper 3-vertex-colouring ϕ in a more clever way. In the next lemma, we show a way to choose ϕ in order to reduce the number of 3s assigned by ℓ to certain sets of edges.

Lemma 5.2.8. *Let G be a graph and let ℓ be a proper $\{0,1,2\}$-labelling of G such that $c_\ell(u) \not\equiv c_\ell(v) \bmod 3$ for every edge $uv \in E(G)$. If H is a (not necessarily connected) spanning d-regular subgraph of G for some $d \geq 1$, then there exists a proper $\{0,1,2\}$-labelling ℓ' of G such that $c_{\ell'}(u) \not\equiv c_{\ell'}(v) \bmod 3$ for every edge $uv \in E(G)$ and that assigns label 0 to at most a third of the edges of $E(H)$. Moreover, for every edge $e \in E(G) \setminus E(H)$, $\ell'(e) = \ell(e)$.*

Proof. We construct the following new labelling: starting from ℓ, add 1 (modulo 3) to all the labels assigned by ℓ to the edges of H. The resulting labelling ℓ_1 is a proper $\{0,1,2\}$-labelling of G such that $c_{\ell_1}(u) \not\equiv c_{\ell_1}(v) \bmod 3$ for every edge $uv \in E(G)$. Indeed, for every $v \in V(G)$, we have $c_{\ell_1}(v) \equiv c_\ell(v) + d \bmod 3$. Thus, if there exist two adjacent vertices $u, v \in V(G)$ such that $c_{\ell_1}(u) \equiv c_{\ell_1}(v) \bmod 3$, then $c_\ell(u) \equiv c_\ell(v) \bmod 3$, a contradiction. We define ℓ_2 in a similar fashion, by adding 1 (modulo 3) to all the labels assigned by ℓ_1 to the edges of H. Similarly, ℓ_2 is proper. Note that, for every edge $e \in E(H)$, we have $\{\ell(e), \ell_1(e), \ell_2(e)\} = \{0,1,2\}$. This implies that at least one of ℓ, ℓ_1, ℓ_2 assigns label 0 to at most a third of the edges of $E(H)$. Finally, since none of the labels of the edges of $E(G) \setminus E(H)$ were changed to obtain ℓ_1 from ℓ and to obtain ℓ_2 from ℓ_1, the last statement of the lemma holds. □

In Lemma 5.2.8, if $d = 2$, then H forms a cycle cover of G. Thus, when H is also an odd unicyclic spanning connected subgraph of G, a particular application of Lemma 5.2.8 in conjunction with the proof of Theorem 5.2.6 gives the following corollary:

Family $\mathcal{F}$	**$\exists$ arbitrarily large $G = (V, E) \in \mathcal{F}$: $mT(G) \geq$**	**$\forall G = (V, E) \in \mathcal{F}$: $mT(G) \leq$**
$\chi(G) = 3$	$\frac{1}{10}\|E\|$	$\frac{\|V\|}{\|E\|}\|E\|$
Cubic other than K_4	$\frac{1}{10}\|E\|$	$\frac{1}{3}\|E\|$
Cactus	$\frac{1}{12}\|E\|$	$\frac{1}{3}\|E\|$
Planar girth $g \geq 5k+1$, $k \geq 7$	$\frac{1}{g(g+1)}\|E\|$	$\frac{2}{k-1}\|E\|$

Table 5.1 – Summary of the results presented in Section 5.3.

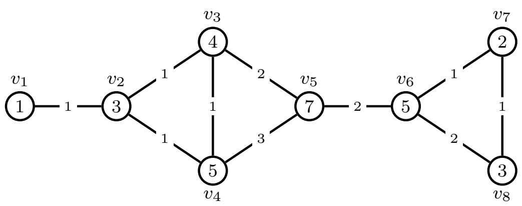
Corollary 5.2.9. *If G is a 3-chromatic Hamiltonian graph of odd order, then $\rho_3(G) \leq 1/3$.*

Another application of Lemma 5.2.8 is for $d = 1$, *i.e.*, H forms a perfect matching. That is, Lemma 5.2.8 in conjunction with the proof of Theorem 5.2.6 can be used, for instance, to prove that class-1 cubic graphs verify Conjecture 2.1.9. Indeed, let G be a class-1 cubic graph, and let M_1, M_2, M_3 be three disjoint perfect matchings of G. We can assume that G is not bipartite, as otherwise Theorem 5.1.1 would apply, and also that G is not K_4 (as it can be checked by hand that $\mathrm{mT}(K_4) = 1$). Thus, by Brooks' Theorem, we get that G is 3-chromatic. Mimicking the proof of Theorem 5.2.6, we can use an odd-length cycle of G to deduce a $\{0, 1, 2\}$-labelling ℓ of G where $c_\ell(u) \not\equiv c_\ell(v) \bmod 3$ for every $uv \in E(G)$. Then, by applying Lemma 5.2.8 on each one of the M_is, we can assume that, for every M_i, at most a third of its edges are assigned label 0 by ℓ. Since the M_i's partition $E(G)$, turning all 0s by ℓ into 3s, we end up with a proper 3-labelling of G where at most a third of the edges are assigned label 3. In Section 5.3, via a different approach, we will actually prove that if G is a cubic graph, then $\rho_3(G) \leq \frac{1}{3}$, *i.e.*, that Conjecture 2.1.9 holds for cubic graphs.

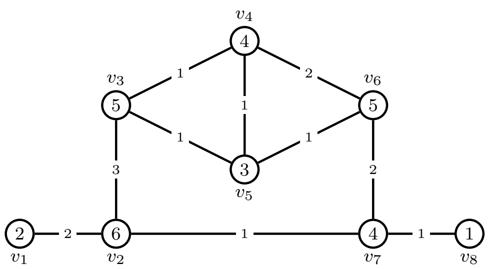
Regarding the proof of Theorem 5.2.6 and the previous arguments, it would be interesting if we could always choose the odd unicyclic subgraph H in such a way that it admits several disjoint perfect matchings, so that Lemma 5.2.8 can be employed to reduce the number of assigned 3s. In the proof of Theorem 5.4.3, we will point out one graph class where this strategy can be employed.

5.3 The parameters mT and ρ_3 for some graph classes

We now use the tools introduced in Section 5.2 to exhibit results on the parameters mT and ρ_3 for some particular classes of 3-chromatic graphs (and beyond). In particular, we prove that, for many classes $\mathcal{F}$ of 3-chromatic graphs, there is no $p \geq 1$ such that $\mathcal{F} \subset \mathcal{G}_{\leq p}$ (*i.e.*, a constant number p of 3s is not sufficient to construct a proper 3-labelling of at least one of the graphs in $\mathcal{F}$). In such cases, we provide upper bounds for $\rho_3(\mathcal{F})$. Our results are summarised in Table 5.1. In Section 5.3.1 we present two graphs, as well as two operations, which will serve throughout the rest of this chapter in constructions that will yield lower bounds on the ρ_3 parameter. Then, in each subsection, we focus on a specific graph family (cubic graphs, planar graphs of big girth and cacti) and provide both an upper and lower bound on the ρ_3 parameter of graphs belonging to the corresponding family.



(a) A_1



(b) A_2

Figure 5.3 – Some proper 3-labellings ℓ of A_1 and A_2 with $\mathrm{nb}_\ell(3) = 1$. The colours by c_ℓ are indicated by integers within the vertices.

5.3.1 Connected graphs needing lots of 3s

As mentioned earlier, we are aware of only two connected graphs for which the parameter ρ_3 is exactly $1/3$, and these are C_3 and C_6 *. A natural question to ask, is whether the bound in Conjecture 2.1.9 is accurate in general, *i.e.*, whether it can be attained by arbitrarily large graphs.

In light of these thoughts, our goal in this subsection is to provide a class of arbitrarily large connected graphs achieving the largest possible ratio ρ_3. Our arguments are based on our notion of weakly induced subgraphs, introduced in Section 5.2. Basically, the idea is to have a connected graph H with $\mathrm{mT}(H) \geq 1$, and to combine p copies $H_1, \dots, H_p$ of H to a single connected graph G so that $\mathrm{mT}(G) \geq p$. To guarantee that $\rho_3(G)$ is large, the main ideas are 1) to choose H so that $|E(H)|$ is as small as possible, and 2) to construct G so that only a few edges join the p copies of H. These two conditions ensure that $|E(G)|$ itself is as small as possible.

We ran computer programs to find graphs H with $\delta(H) = 1$, $\mathrm{mT}(H) \geq 1$, and with the fewest edges possible. It turns out that the smallest such graphs have 10 edges. Two such graphs, which we call A_1 and A_2, are depicted in Figure 5.3. These two graphs will be used throughout the rest of this chapter. The two graphs A_1 and A_2 will allow us to prove several lower bounds on ρ_3 for various graph classes, so, let us formally establish that they do have the desired property.

Lemma 5.3.1. $\mathrm{mT}(A_1) = 1$.

Proof. A proper 3-labelling ℓ of A_1 with $\mathrm{nb}_\ell(3) = 1$ is depicted in Figure 5.3(a), which shows that $\mathrm{mT}(A_1) \leq 1$. We now prove that $\mathrm{mT}(A_1) > 0$, *i.e.*, that there is no proper 2-labelling of A_1. Towards a contradiction, assume a proper 2-labelling ℓ of A_1 exists.

By Observation 2.1.5, we have $\ell(v_6v_7) \neq \ell(v_6v_8)$. Also, since ℓ is a 2-labelling, we have $c_\ell(v_5) \in \{3, 4, 5, 6\}$. We distinguish the following cases:

- **Case 1:** $c_\ell(v_5) = 3$. Then, $\ell(v_3v_5) = \ell(v_4v_5) = \ell(v_5v_6) = 1$, and so, $\{c_\ell(v_3), c_\ell(v_4)\} = \{4, 5\}$. Assume, w.l.o.g., that $c_\ell(v_3) = 4$ and $c_\ell(v_4) = 5$. It follows that $\ell(v_2v_3) = 1$ and $\ell(v_2v_4) = \ell(v_3v_4) = 2$, and thus, $c_\ell(v_2) \in \{4, 5\} = \{c_\ell(v_3), c_\ell(v_4)\}$, which contradicts that ℓ is proper.
- **Case 2:** $c_\ell(v_5) = 4$. Then, v_5 has exactly one incident edge labelled 2. First, assume that $\ell(v_5v_6) = 2$. It follows that $\ell(v_3v_5) = \ell(v_4v_5) = 1$, and thus, $\{c_\ell(v_3), c_\ell(v_4)\} = \{3, 5\}$. Assume, w.l.o.g, that $c_\ell(v_3) = 3$ and $c_\ell(v_4) = 5$. Since $c_\ell(v_3) = 3$, we have that $\ell(v_2v_3) =$

*. Any disjoint union of C_3's and C_6's reaches that value. This is why Conjecture 2.1.9 focuses on connected graphs.

$\ell(v_3v_4) = 1$, and thus, $c_\ell(v_4) \leq 4$, a contradiction. Second, assume that $\ell(v_5v_6) = 1$. Since $\{\ell(v_6v_7), \ell(v_6v_8)\} = \{1, 2\}$, we have $c_\ell(v_6) = 4 = c_\ell(v_5)$, a contradiction.

For the next two cases, let $A_1' = A_1 - \{v_7v_8\}$ and observe that A_1' is quasi 3-regular.

— **Case 3:** $c_\ell(v_5) = 5$. Then, by Lemma 5.1.3, the 2-labelling $\ell' = \mathrm{sw}(\ell|_{A_1'}, \sigma_{1\leftrightarrow 2})$ is also proper for A_1'. Moreover, recall that $\{\ell'(v_6v_7), \ell'(v_6v_8)\} = \{1, 2\}$. It follows that ℓ' can be extended to a proper 2-labelling ℓ'' of A_1 by setting $\ell''(v_7v_8) = 1$. But then, $c_{\ell''}(v_5) = 4$, and we get a contradiction to **Case 2** above.

— **Case 4:** $c_\ell(v_5) = 6$. Similarly to the previous case, the 2-labelling $\ell' = \mathrm{sw}(\ell|_{A'}, \sigma_{1\leftrightarrow 2})$ is proper for A_1' and it can be extended to a proper 2-labelling ℓ'' of A_1 by setting $\ell''(v_7v_8) = 1$. But then, $c_{\ell''}(v_5) = 3$, and we get a contradiction to **Case 1** above.

□

Lemma 5.3.2. $\mathrm{mT}(A_2) = 1$.

Proof. A proper 3-labelling ℓ of A_2 with $\mathrm{nb}_\ell(3) = 1$ is depicted in Figure 5.3(b). Thus, $\mathrm{mT}(A_2) \leq 1$. Let us prove now that $\mathrm{mT}(A_2) > 0$, *i.e.*, that there is no proper 2-labelling of A_2. Towards a contradiction, assume a proper 2-labelling ℓ of A_2 exists.

Since ℓ is a 2-labelling, we have $c_\ell(v_3) \in \{3, 4, 5, 6\}$. We distinguish the following cases:

— **Case 1:** $c_\ell(v_3) = 3$. Then, $\ell(v_2v_3) = \ell(v_3v_4) = \ell(v_3v_5) = 1$, and so, $\{c_\ell(v_4), c_\ell(v_5)\} = \{4, 5\}$. Assume, w.l.o.g., that $c_\ell(v_4) = 4$ and $c_\ell(v_5) = 5$. It follows that $\ell(v_5v_4) = \ell(v_5v_6) = 2$ and $\ell(v_4v_6) = 1$, and thus, $c_\ell(v_6) \in \{4, 5\} = \{c_\ell(v_4), c_\ell(v_5)\}$, which contradicts that ℓ is proper.

— **Case 2:** $c_\ell(v_3) = 4$. Then, v_3 has exactly one incident edge labelled 2. First, assume that $\ell(v_3v_2) = 2$. It follows that $\ell(v_3v_4) = \ell(v_3v_5) = 1$, and thus, $\{c_\ell(v_4), c_\ell(v_5)\} = \{3, 5\}$. Assume, w.l.o.g., that $c_\ell(v_4) = 3$ and $c_\ell(v_5) = 5$. Since $c_\ell(v_4) = 3$, we have that $\ell(v_4v_5) = 1$, and thus, $c_\ell(v_5) \leq 4$, a contradiction. Then, assume, w.l.o.g., that $\ell(v_3v_5) = 2$ (and $\ell(v_3v_2) = \ell(v_3v_4) = 1$). It follows that $c_\ell(v_5) \in \{5, 6\}$ and $c_\ell(v_4) \in \{3, 5\}$. If $c_\ell(v_4) = 5$, then $c_\ell(v_5) = 6$. This implies that $\ell(v_4v_6) = \ell(v_5v_6) = 2$, and thus, $c_\ell(v_6) \in \{5, 6\} = \{c_\ell(v_4), c_\ell(v_5)\}$, a contradiction. Otherwise, $c_\ell(v_4) = 3$, and so, $c_\ell(v_5) = 5$ and $c_\ell(v_6) = 4$. Hence, $\ell(v_6v_7) = \ell(v_3v_2) = 1$, $c_\ell(v_2), c_\ell(v_7) \in \{3, 5\}$ (because $c_\ell(v_3) = c_\ell(v_6) = 4$). We now get a contradiction no matter how v_1v_2, v_2v_7, and v_7v_8 are labelled, as either $c_\ell(v_2) = c_\ell(v_7)$ or $4 \in \{c_\ell(v_2), c_\ell(v_7)\}$.

— **Case 3:** $c_\ell(v_3) = 5$. Then, by Lemma 5.1.3, the 2-labelling $\ell' = \mathrm{sw}(\ell, \sigma_{1\leftrightarrow 2})$ is also proper (note that A_2 is quasi 3-regular). Since $c_{\ell'}(v_3) = 4$, we get a contradiction to **Case 2** above.

— **Case 4:** $c_\ell(v_3) = 6$. Then, by Lemma 5.1.3, the 2-labelling $\ell' = \mathrm{sw}(\ell, \sigma_{1\leftrightarrow 2})$ is also proper. Since $c_{\ell'}(v_3) = 3$, we get a contradiction to **Case 1** above.

□

Through the next constructions, A_1 and A_2 will be used to build arbitrarily large connected graphs with large ρ_3 and particular properties. Let G be a graph. Given a graph H with at least two distinct vertices of degree 1, we define *H-augmenting* an edge uv of G by the following operations:

1. deleting uv from G;
2. adding a copy of H to G;
3. identifying u and any degree-1 vertex of H, and identifying v and any other degree-1 vertex of H.

Analogously, assuming H has at least one vertex of degree 1, by *H-attaching* a pending edge uv of G, where v has degree 1, we mean the following:

1. deleting v from G;
2. adding a copy of H to G;
3. identifying u and any degree-1 vertex of H.

The next lemma illustrates how these two operations can be used:

Lemma 5.3.3. *Let G be a nice graph and let H be a graph with at least two vertices of degree 1 (at least one vertex of degree 1, respectively). Let G' be the graph obtained by H-augmenting (H-attaching, respectively) p distinct edges (pending edges, respectively) of G (where $1 \leq p \leq |E(G)|$). Then,* $\mathrm{mT}(G') \geq p \cdot \mathrm{mT}(H)$.

Proof. This follows from Lemma 5.2.2 since G' contains p copies $H_1, \dots, H_p$ of H as pairwise disjoint weakly induced subgraphs. □

The following theorem can be deduced from Lemma 5.3.3 since both graphs A_1 and A_2 have degree-1 vertices, verify the properties of Lemmata 5.3.1 and 5.3.2, and have 10 edges.

Theorem 5.3.4. *There exist arbitrarily large connected graphs G with $\rho_3(G) \geq 1/10$.*

Proof. Let $p \geq 1$ be fixed. We construct a connected graph G with $10p$ edges such that $\mathrm{nb}_\ell(3) \geq p$ for any proper 3-labelling ℓ of G, which implies that $\rho_3(G) \geq 1/10$. One possible construction (using A_2) is as follows. Start from any connected graph with p edges, and A_2-augment all the p edges to get G. Then G has the claimed properties due to Lemmata 5.3.2 and 5.3.3. □

5.3.2 Bounds for connected cubic graphs

Recall that, given a cubic graph G, it is NP-complete to decide whether $\chi_\Sigma(G) \leq 2$ (see [58]). Then, a natural question to ask is whether they always admit proper 3-labellings assigning only a limited number of 3s. We prove that there is actually no $p \geq 1$ such that the class of all cubic graphs lies in $\mathcal{G}_{\leq p}$. In contrast, we verify Conjecture 2.1.9 for this class of graphs.

First off, we note that the construction in the proof of Theorem 5.3.4 can be modified slightly to reach the same conclusion for cubic graphs.

Theorem 5.3.5. *There exist arbitrarily large connected cubic graphs G with $\rho_3(G) \geq 1/10$.*

Proof. This follows from applying the same construction as in the proof of Theorem 5.3.4, but starting from a connected cubic graph with p edges, where p is a multiple of 3. In particular, note that A_2 is quasi 3-regular with exactly two degree-1 vertices (the ones that are used during the A_2-augmentations), which implies that the resulting graph G is cubic. □

Note that, through playing with A_2-augmentations and the starting graph, we can go a bit beyond Theorem 5.3.5. For instance, since A_2 has exactly two cut vertices and each one is adjacent to one of its two degree-1 vertices, it can be checked that, performing the construction described in the proof of Theorem 5.3.5 starting from 2-connected cubic graphs, yields arbitrarily large 2-connected cubic graphs G with $\rho_3(G) \geq 1/10$.

Regarding upper bounds, we prove that the parameter ρ_3 cannot exceed the $1/3$ barrier in cubic graphs. In other words, we prove Conjecture 2.1.9 for these graphs.

Theorem 5.3.6. *If G is a cubic graph, then $\rho_3(G) \leq 1/3$.*

Proof. We can assume that G is connected. Also, we can assume that G is neither K_4 (in which case the claim can be verified by hand) nor bipartite (due to Theorem 5.1.1). Thus, by Brooks' Theorem, we know that G is 3-chromatic. Recall that $|E(G)| = \frac{3}{2}|V(G)|$.

Let us now mimic the proof of Theorem 5.2.6 to get a proper 3-labelling ℓ of G such that, for every edge $e \in E(G) \setminus E(H)$ (where, recall, H is an odd unicyclic spanning connected subgraph of G), we have $\ell(e) = 2$. This means that only the edges of H can be labelled 1 or 3 by ℓ. If $\mathrm{nb}_\ell(3) \leq \frac{1}{2}|E(H)|$, then the result follows since $|E(H)| = \frac{2}{3}|E(G)|$. So, assume now that $\mathrm{nb}_\ell(3) > \frac{1}{2}|E(H)|$, and hence, $\mathrm{nb}_\ell(1) < \frac{1}{2}|E(H)|$. Since G is regular, by Lemma 5.1.2, the 3-labelling $\ell' = \mathrm{sw}(\ell, \sigma_{1\leftrightarrow 3})$ of G is also proper. Since only the edges of H are labelled 1 or 3 by ℓ, we deduce that $\mathrm{nb}_{\ell'}(3) = \mathrm{nb}_\ell(1) < \frac{1}{2}|E(H)| = \frac{1}{3}|E(G)|$, and the result follows. ☐

5.3.3 Bounds for connected planar graphs with large girth

Recall that the *girth* $g(G)$ of a graph G is the length of a shortest cycle of G. For any $g \geq 3$, we denote by $\mathcal{P}_g$ the class of planar graphs with girth at least g. Note, for instance, that $\mathcal{P}_3$ is the class of all (simple) planar graphs, and that $\mathcal{P}_4$ is the class of all triangle-free planar graphs. Recall that the girth of a tree is set to ∞, since it has no cycle.

To date, it is still unknown whether planar graphs verify the 1-2-3 Conjecture, which makes the study of the parameters mT and ρ_3 adventurous for this class of graphs. Something we can state, however, is that there is no $p \geq 1$ such that planar graphs lie in $\mathcal{G}_{\leq p}$. Indeed, since the graphs A_1 and A_2 are planar, this can be established by the construction in the proof of Theorem 5.3.4 (or from that of Theorem 5.3.5 to additionally get a cubic graph assumption), by performing it from planar starting graphs.

Theorem 5.3.7. *There exist arbitrarily large connected planar graphs G with $\rho_3(G) \geq 1/10$.*

To go further, we can consider planar graphs with large girth. Indeed, as established by Grötzsch's Theorem, triangle-free planar graphs are 3-colourable, which means that they verify the 1-2-3 Conjecture (recall Theorem 2.2.1). In what follows, we prove two main results. First, we prove that, for every $g \geq 3$, there is no $p \geq 1$ such that $\mathcal{P}_g \subseteq \mathcal{G}_{\leq p}$. Second, we prove that, as the girth $g(G)$ of a planar graph G grows, the ratio $\rho_3(G)$ decreases. As a side result, we prove Conjecture 2.1.9 for planar graphs with girth at least 36.

In order to prove the first result above, note that we cannot use the graphs A_1 and A_2 introduced previously, as they contain triangles. Instead, we provide another construction, yielding, for any $g \geq 3$, a planar graph S_g with girth g. Start from S_g being the cycle $C_g = (v_0, \dots, v_{g-1}, v_0)$ on g vertices. Then, for each $i \in \{0, \dots, g-1\}$, add a new vertex $u_{i,1}$ and the edge $v_i u_{i,1}$ to S_g. Then, for every $i \in \{1, \dots, g-1\}$, add a cycle $B_i = (u_{i,1}, u_{i,2}, \dots, u_{i,g}, u_{i,1})$ to S_g, where $u_{i,2}, \dots, u_{i,g}$ are new vertices. Finally, let $u_{0,1}$ be the *root* of S_g. See Figure 5.4 for an illustration of S_3 and S_g. It is clear that all the cycles of S_g have length g, and thus, $g(S_g) = g$. Moreover, S_g is clearly planar, and $|E(S_g)| = g^2 + g$.

Note that S_g is bipartite whenever g is even. Since $\delta(S_g) = 1$, in such cases we have $\mathrm{mT}(S_g) = 0$ by Theorem 5.1.1. When $g \equiv 1 \bmod 4$, it can be checked (for instance, by using some of the arguments in the proof of upcoming Lemma 5.3.8) that S_g admits proper 2-labellings, and thus, we have $\mathrm{mT}(S_g) = 0$ in those cases as well. The main point for considering this construction is for the last possible values of g, the values where $g \equiv 3 \bmod 4$, for which the following is verified:

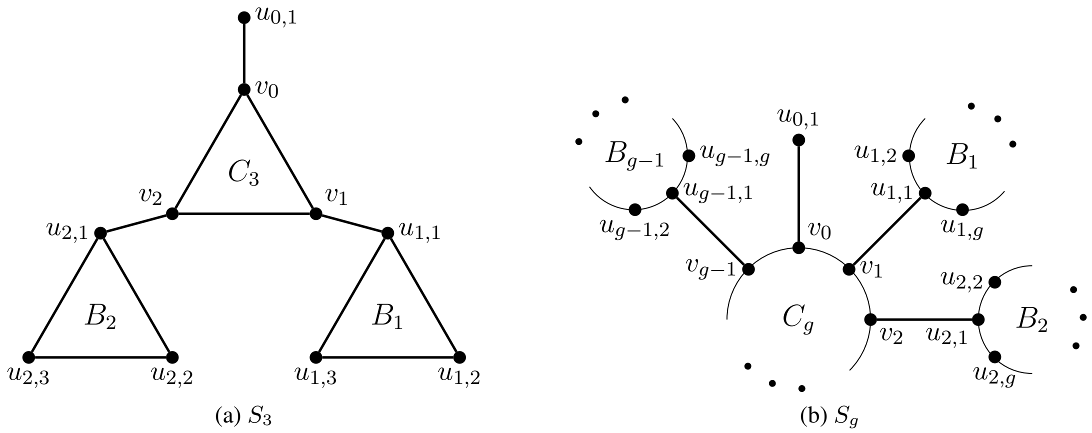


Figure 5.4 – The planar graphs S_3 (left) and S_g (right) of girth 3 and g, respectively.

Lemma 5.3.8. *For every $g \geq 3$ with $g \equiv 3 \bmod 4$, we have* $\mathrm{mT}(S_g) = 1$.

Proof. We begin by showing that a proper 2-labelling of S_g must have specific properties. In what follows, for every $i \in \{1, \ldots, g-1\}$, we denote by H_i the subgraph of S_g induced by $V(B_i) \cup \{v_i\}$.

Claim 5.3.9. *Let $i \in \{1, \ldots, g-1\}$. By any proper 2-labelling ℓ of H_i, we have $\ell(u_{i,1}u_{i,2}) \neq \ell(u_{i,g}u_{i,1})$, and thus, $c_\ell(u_{i,1}) = \ell(u_{i,1}v_i) + 3$. Furthermore, such a proper 2-labelling exists.*

Proof of the claim. The first part of the claim follows from Observation 2.1.5. Indeed, since $g \equiv 3 \bmod 4$, it follows that we must have $\ell(u_{i,1}u_{i,2}) \neq \ell(u_{i,3}u_{i,4}) \neq \ldots \neq \ell(u_{i,g}u_{i,1})$. Now, it is easy to check that the following is a proper 2-labelling ℓ of H_i. Start by setting $\ell(u_{i,1}u_{i,2}) = 2$. Then, continue from $u_{i,2}u_{i,3}$ and, following the edges of B_i until reaching $u_{i,g}u_{i,1}$, assign labels $1, 1, 2, 2, 1, 1, 2, \ldots, 2, 1, 1, 2, 2, 1, 1$. The edge $u_{i,1}v_i$ can then be assigned any label in $\{1, 2\}$. $\diamond$

Assume that there exists a proper 2-labelling ℓ of S_g, and let $\{\alpha, \beta\}$ be a permutation of $\{1, 2\}$. We define the set

$$J = \{j \in \{0, \ldots, g-1\} : \ell(v_{j-1}v_j) \neq \ell(v_jv_{j+1})\},$$

where, here and in what follows, indices are taken modulo g. Observe that $|J| \equiv 0 \bmod 2$ and that $J \neq \emptyset$. Indeed, assume that $J = \emptyset$. Then, we would have that all the edges of C_g receive the same label α or β. Since ℓ is proper, it must be that $\ell(v_iu_{i,1}) \neq \ell(v_{i+1}u_{i+1,1})$ for all $0 \leq i \leq g-1$. This is a contradiction since g is odd and ℓ is a 2-labelling. Let $j \in J$ such that $j \geq 1$ (the vertex v_j exists since $|J| \equiv 0 \bmod 2$ and $J \neq \emptyset$). We have that $\ell(v_{j-1}v_j) \neq \ell(v_jv_{j+1})$, which implies that $\ell(v_{j-1}v_j) + \ell(v_jv_{j+1}) = 3$. It follows that $c_\ell(v_j) = \ell(v_ju_{j,1}) + 3$. Note now that the labelling $\ell_j = \ell|_{H_j}$ is a proper 2-labelling of H_j (since $d_{H_j}(v_j) = 1$). Therefore, by Claim 5.3.9, it follows that $c_\ell(u_{j,1}) = c_{\ell_j}(u_{j,1}) = \ell_j(u_{j,1}v_j) + 3 = \ell(u_{j,1}v_j) + 3 = c_\ell(v_j)$, a contradiction.

So far, we have proved that $\mathrm{mT}(S_g) \geq 1$. In order to show that $\mathrm{mT}(S_g) = 1$, it suffices to provide a proper 3-labelling ℓ of S_g such that $\mathrm{nb}_\ell(3) = 1$. We construct one such labelling as follows. For every $i \in \{1, \ldots, g-1\}$, we label the subgraph B_i following the 2-labelling scheme provided in Claim 5.3.9. Then, we set $\ell(v_0v_1) = 3$ and, for every edge $e \in E(C_g) \setminus \{v_0v_1\}$, we set $\ell(e) = 1$. Finally, for the edges of the form $v_iu_{i,1}$ $(0 \leq i \leq g-1)$, we set $\ell(v_0u_{0,1}) = 1$, $\ell(v_1u_{1,1}) = 2$, $\ell(v_2u_{2,1}) = 1, \ldots, \ell(v_{g-2}u_{g-2,1}) = 2$, $\ell(v_{g-1}u_{g-1,1}) = 1$. It is clear that $c_\ell(v_0) = 5$, and $c_\ell(v_1) = 6$, while the colours of the vertices of the rest of the cycle C_g alternate between 3 and 4. Moreover, for all $2 \leq i \leq g-1$, if $c_\ell(v_i) = 3$, then $c_\ell(u_{i,1}) = 4$, and if $c_\ell(v_i) = 4$, then $c_\ell(u_{i,1}) = 5$ (by Claim 5.3.9). Thus, ℓ is a proper 3-labelling that assigns label 3 to only one edge of S_g. □

We are now ready to prove our lower bound.

Theorem 5.3.10. *For every $g' \geq 3$, there exist arbitrarily large connected planar graphs G with $g(G) \geq g'$ and $\rho_3(G) \geq \frac{1}{g^2+g}$, where g is the smallest natural number such that $g \geq g'$ and $g \equiv 3 \bmod 4$.*

Proof. For any integer $p \geq 1$, denote by G the graph obtained from p disjoint copies $H_1, \dots, H_p$ of S_g by identifying their roots to a single vertex. Clearly, G is planar and has girth $g \geq g'$. Furthermore, G clearly contains p copies of S_g as pairwise disjoint weakly induced subgraphs. Then, Lemma 5.2.2 implies that $\mathrm{mT}(G) \geq p \cdot \mathrm{mT}(S_g)$, and $p \cdot \mathrm{mT}(S_g) = p$ by Lemma 5.3.8. Since G has $p|E(S_g)| = p(g^2+g)$ edges, the result follows. Moreover, these arguments apply for any value of p, and so, G can be as large as desired. □

It is not too complicated to check that our construction in the proof of Theorem 5.3.10 yields planar graphs G of girth g satisfying $\mathrm{mT}(G) = |E(G)|/(g^2+g)$ (when $g \equiv 3 \bmod 4$). Note also that the graph G constructed in the proof of Theorem 5.3.10 does not have girth g' when $g' \not\equiv 3 \bmod 4$. In this case, to obtain a similar result for a graph of girth g', we can additionally identify a single vertex of a new cycle of length g' to the same single vertex as the roots of the p copies of S_g in the previous proof.

We now proceed to prove that $\rho_3(G) \leq \frac{2}{k-1}$ for any nice planar graph G of girth $g \geq 5k+1$, when $k \geq 7$. In other words, the bigger the girth of a planar graph G, the smaller $\rho_3(G)$ gets.

The following theorem from [101] is one of the main tools we use to prove this result. For any $k \geq 1$, a *k-thread* in a graph G is a path $(u_1, \dots, u_{k+2})$, where the k inner vertices $u_2, \dots, u_{k+1}$ all have degree 2 in G.

Theorem 5.3.11 ([101]). *For any integer $k \geq 1$, every planar graph with minimum degree at least 2 and girth at least $5k+1$ contains a k-thread.*

We can now proceed with the main theorem.

Theorem 5.3.12. *Let $k \geq 7$. If G is a nice planar graph with $g(G) \geq 5k+1$, then $\rho_3(G) \leq \frac{2}{k-1}$.*

Proof. Throughout this proof, we set $g = g(G)$. The proof is by induction on the order of G. The base case is when $|V(G)| = 3$. In that case, G must be a path of length 2 (due to the girth assumption), and the claim is clearly true. So let us focus on proving the general case.

We can assume that G is connected. If G is a tree, then $\chi_\Sigma(G) \leq 2$ and we have $\rho_3(G) = 0$. So, from now on, we may assume that G is not a tree. We can also assume that G has no vertex v to which is attached a pending tree T_v that is not a star with center v. Indeed, if such a T_v exists, then we can find a vertex $u \in V(T_v) \setminus \{v\}$ whose all neighbours $u_1, \dots, u_x$ but one are degree-1 vertices. Since G is not a tree, the graph $G' = G - \{u_1, \dots, u_x\}$ is clearly a nice planar graph with girth g, admitting, by the inductive hypothesis, a proper 3-labelling attesting that $\rho_3(G') \leq \frac{2}{k-1}$. Lemma 5.2.5 tells us that such a labelling can be extended to one of G.

Let G^- be the graph obtained from G by removing all vertices of degree 1. Note that removing vertices of degree 1 from G can neither decrease its girth nor result in a tree. Since G has girth $g \geq 5k+1$ and does not contain any cut vertex $v \in V(G)$ as described above, the graph G^-

has minimum degree 2. By Theorem 5.3.11, G^- contains a k-thread P. Let $u_1, \dots, u_{k+2}$ be the vertices of P, where $d_{G^-}(u_i) = 2$ for all $2 \le i \le k+1$. Thus, the vertices of P exist in G except that each of the vertices u_i (for $2 \le i \le k+1$) may be adjacent to some vertices of degree 1 in addition to their adjacencies in G^-. Let G' be the graph obtained from G by removing the vertices $u_3, \dots, u_k$ and all of their neighbours that have degree 1 in G. Note that G' might contain up to two connected components. In case G' has exactly two connected components, then, due to a previous assumption, none of these can be a tree, which implies that G' is nice. If G' is connected, then, because it has at least two edges (u_1u_2 and $u_{k+1}u_{k+2}$), it must be nice. Furthermore, in both cases, the girth of G' is at least that of G. Then, by combining the inductive hypothesis and the fact that $\rho_3(T) = 0$ for every nice tree T, we deduce that $\rho_3(G') \le \frac{2}{k-1}$.

To obtain a proper 3-labelling ℓ of G such that $\rho_3(G) \le \frac{2}{k-1}$, we extend a proper 3-labelling ℓ' of G' corresponding to $\rho_3(G') \le \frac{2}{k-1}$, as follows. First, for each edge incident to a vertex of degree 1 that we have removed, label it with 1. Recall that none of these vertices of degree 1 can, later on, be in conflict with their neighbour since they have degree 1. Now, for each $2 \le j \le k-2$, in increasing order of j, label the edge u_ju_{j+1} with 1 or 2, so that the resulting colour of u_j does not conflict with the colour of u_{j-1}. Finally, label the edges $u_{k-1}u_k$ and u_ku_{k+1} with 1, 2 or 3, so that the resulting colour of u_{k-1} does not conflict with that of u_{k-2}, the resulting colour of u_k does not conflict with that of u_{k-1} nor with that of u_{k+1}, and the resulting colour of u_{k+1} does not conflict with that of u_{k+2}. Indeed, this is possible since there exist at least two distinct labels $\{\alpha, \beta\}$ ($\{\alpha', \beta'\}$, respectively) in $\{1, 2, 3\}$ for $u_{k-1}u_k$ (u_ku_{k+1}, respectively) such that the colour of u_{k-1} (u_{k+1}, respectively) is not in conflict with that of u_{k-2} (u_{k+2}, respectively). Thus, w.l.o.g., choose α and α' for the labels of $u_{k-1}u_k$ and u_ku_{k+1}, respectively. If the colour of u_k does not conflict with that of u_{k-1} nor with that of u_{k+1}, then we are done. If the colour of u_k conflicts with both that of u_{k-1} and that of u_{k+1}, then it suffices to change both the labels of $u_{k-1}u_k$ and u_ku_{k+1} to β and β', respectively. Lastly, w.l.o.g., if the colour of u_k only conflicts with that of u_{k-1}, then it suffices to change the label of u_ku_{k+1} to β'. The resulting labelling ℓ of G is thus proper. Moreover, $|E(G) \setminus E(G')| \ge k-1$ and ℓ assigns label 3 to at most two more edges than ℓ', and so, the result follows. $\square$

Corollary 5.3.13. *If G be a planar graph of girth $g(G) \ge 36$, then $\rho_3(G) \le \frac{1}{3}$.*

5.3.4 Bounds for connected cacti

Recall that a *cactus* is a graph in which every edge is contained in at most one simple cycle. Note that trees are also cacti since they do not contain cycles.

First off, note that the graphs S_g introduced in Section 5.3.3, and those we have constructed from them in the proof of Theorem 5.3.10, are all cacti (all of their cycles are actually disjoint). Since the smallest graph S_g is S_3, which has 12 edges, the proof of that theorem implies the following.

Theorem 5.3.14. *There exist arbitrarily large connected cacti G with $\rho_3(G) \ge 1/12$.*

We now focus on the upper bound. We actually end up proving Conjecture 2.1.9 for cacti. Note that cacti are planar graphs and they are 3-chromatic. Thus we know already that if G is a cactus, then $\chi_\Sigma(G) \le 3$. Moreover, if G is of big enough girth, then it also verifies Conjecture 2.1.9 thanks to Corollary 5.3.13. So in some sense, the following proof may appear as unimportant, especially considering it is rather technical and long. Nevertheless, we stress the fact that not much is known on the behaviour of proper labellings of graphs that are not bipartite, apart the fact that 3-chromatic graphs verify the 1-2-3 Conjecture. In our opinion, this gives enough value to the following proof to merit its inclusion in this thesis.

Theorem 5.3.15. *If G is a nice cactus, then $\rho_3(G) \leq 1/3$.*

Proof. The proof is done by induction on $|V(G)|$. Since the claim is clearly true when G has only three vertices, let us consider the general case. Clearly, we can assume that G is connected (as otherwise we could use the inductive hypothesis on each connected component), is not a tree (since $\mathrm{mT}(T) = 0$ for every nice tree T), is not bipartite (by Theorem 5.1.1), and is not a cycle (recall Theorem 4.1.6).

Throughout this proof, for readability reasons, we say that a proper 3-labelling is *good* if it assigns label 3 to at most a third of the edges of the labelled graph. We first prove that if G has some specific properties, then we can remove some vertices from G, resulting in a nice cactus G' that is smaller than G, and extend a good labelling ℓ' of G', obtained by induction, into a good labelling ℓ of G, thus proving the statement for G. It can then be assumed that G does not have these properties, which will simplify its structure and allow us to prove the final inductive step.

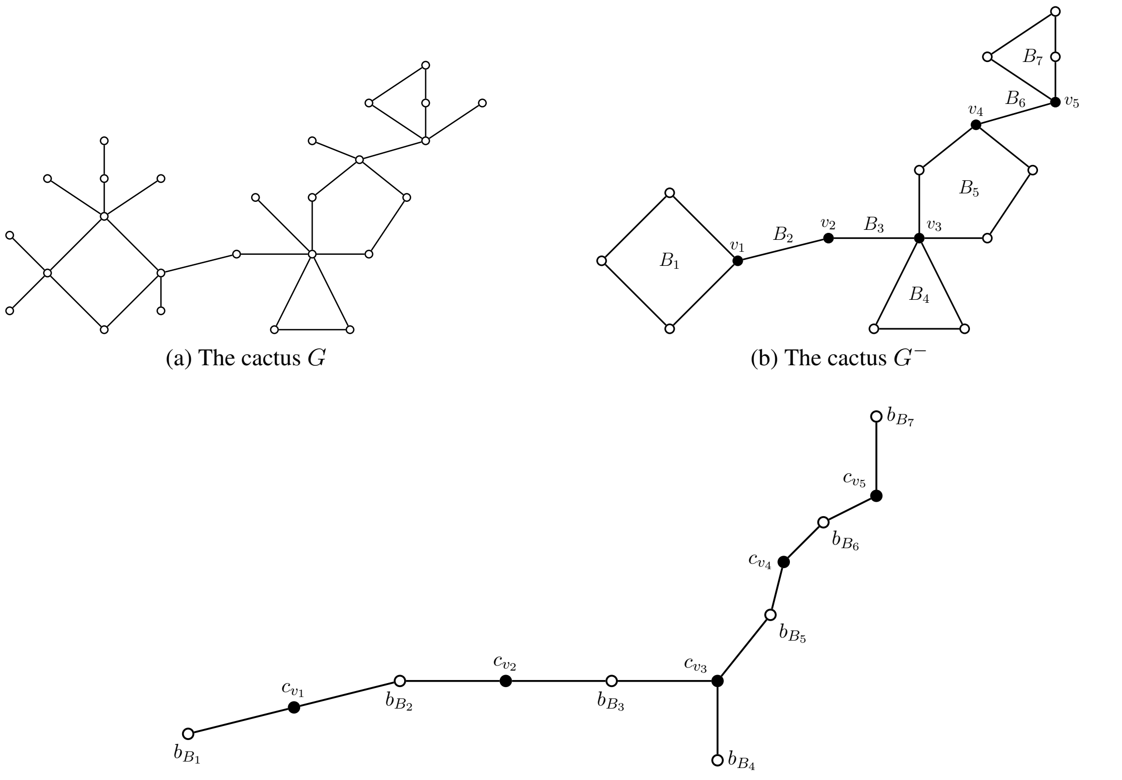
Let us state a few more remarks. Let ℓ be an extension of ℓ' that assigns labels from $\{1, 2\}$ to the edges of G that are not in G'. If this ℓ is proper, then note that it is also good. Similarly, if ℓ assigns label 3 to at most a third of the edges of G that are not in G' and ℓ is proper, then it is also good.

We start by analysing certain cycles of G. To define those cycles, let us consider the following terminology (see Figure 5.5 for an accompanying illustration). We denote by G^- the cactus obtained from G by repeatedly deleting vertices of degree 1 until the remaining graph has minimum degree 2. Since G contains cycles, note that G^- is not empty. We now consider the block graph $B(G^-)$ of G^-, which is defined as follows [59]. A *block* of G^- is a maximal 2-connected subgraph of G^-. The *block graph* $B(G^-)$ is the tree having a *block vertex* b_B for every block B of G^-, a vertex c_v for every cut vertex v of G^-, and in which two vertices b_B and c_v are joined by an edge if and only if B contains v in G^-. Note that $B(G^-)$ is not empty since G^- has at least one cycle, and, due to how G^- was obtained from G, note that all the leaves of $B(G^-)$ are block vertices corresponding to cycles in G^-. In what follows, we study structures around *end-cycles*, where an end-cycle C of G refers to a cycle of G^-, which corresponds to a leaf b_C of $B(G^-)$. In G^-, every vertex of an end-cycle C has degree 2, except for one, which we denote by r and call the *root* of C, while its other vertices are the *inner vertices* of C. Note that end-cycles are better defined as soon as G has at least two cycles. In case G has only one cycle C, then we consider C as an end-cycle, its root being any of its vertices of degree more than 2 in G (at least one exists since G is not a cycle).

In what follows, we consider any end-cycle C of G. We first investigate properties of pending trees attached to the vertices of C. For every vertex v of C, we define T_v as the pending tree rooted at v in G. Note that there might be no edges in such a T_v, *i.e.*, we can have $V(T_v) = \{v\}$. We implicitly assume that every T_v comes with the natural (virtual) orientation of its edges from the root (v) to the leaves. Also, we say that T_v is *inner* if v is indeed an inner vertex of C.

Claim 5.3.16. *If some T_v has edges and is not a star, then there is a good labelling of G.*

Proof of the claim. Let us consider a deepest (*i.e.*, farthest from v) vertex u of T_v, where all of its $x \geq 1$ children are leaves. Since T_v is not a star, we have $u \neq v$. Then, the graph G' obtained from



(a) The cactus G

(b) The cactus G^-

(c) The block graph $B(G^-)$

Figure 5.5 – An example of a cactus G, with the corresponding cactus G^- and the block graph $B(G^-)$, as they are introduced in the proof of Theorem 5.3.15. The black vertices of G^- are cut vertices of G^-. Observe that the leaves b_{B_1}, b_{B_4}, and b_{B_7} of $B(G^-)$, correspond exactly to the cycles B_1, B_4, and B_7 of G (and of G^-), which are considered as end-cycles of G, while B_5 is not considered as an end-cycle of G since b_{B_5} is not a leaf of $B(G^-)$. Clearly, b_{B_1} is at distance 10 from b_{B_7}, while b_{B_4} is at distance 6 from both b_{B_1} and b_{B_7} in $B(G^-)$.

G by removing all of these x leaves is a nice cactus (due to the presence of the cycle C) in which u has degree 1. Thus, G' admits a good labelling by the inductive hypothesis. Lemma 5.2.5 tells us that this good labelling of G' can be extended to one of G. $\diamond$

Claim 5.3.17. *If some inner T_v is a star with at least two edges, then there is a good labelling of G.*

Proof of the claim. Let G' be the graph obtained from G by removing two leaves u, u' of T_v. Clearly, G' is a cactus, and G' is nice due to the presence of C. By the inductive hypothesis, there is a good labelling of G'. To obtain one of G, it suffices to extend this labelling to vu and vu' by assigning labels 1 and 2 in such a way that no colour conflict arises. Recall that, by a k-labelling of a nice graph, a vertex of degree 1 cannot be involved in a colour conflict with its neighbour. Then, it suffices to label vu and vu' so that no colour conflict arises between v and its two neighbours in C. Note that there are three different ways to label edges vu and vu' (assigning label 1 twice, assigning 2 twice, or assigning both 1 and 2 once). Under these labellings, the vertex v can take three different colours, while it has two neighbours in C. Hence, at least one labelling for the two edges extends the labelling of G' to a good labelling of G. $\diamond$

Thus, in C, any inner T_v can be assumed to have at most one edge.

Claim 5.3.18. *If C has length at least 4 and some inner T_v has an edge, then there is a good labelling of G.*

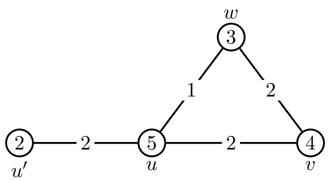
Proof of the claim. Assume $C = (v_0, v_1, \dots, v_{n-1}, v_0)$, where $v_0 = r$ is the root of C and $n \geq 4$. By Claims 5.3.16 and 5.3.17, each T_{v_i} (where $i \in \{1, \dots, n-1\}$) has at most one edge.

Assume first that there is an $i \in \{2, \dots, n-2\}$ such that T_{v_i} has an edge $v_i u$. Let G' be the graph obtained from G by removing u and v_i. Clearly, G' is a cactus with at least two edges (v_0v_1 and $v_{n-1}v_0$), so it is nice. By the inductive hypothesis, there is a good labelling of G', which we want to extend to one of G. To that aim, we have to label the three edges $v_iu, v_iv_{i-1}, v_iv_{i+1}$ (where, here and in what follows, indices are taken modulo n) so that no colour conflict arises, and label 3 is assigned at most once. First, we assign 1 or 2 to v_iv_{i-1} so that v_{i-1} does not get in conflict with v_{i-2}. Second, we assign 1 or 2 to v_iv_{i+1} so that v_{i+1} does not get in conflict with v_{i+2}. Third, we assign 1, 2 or 3 to v_iu so that v_i gets in conflict with neither v_{i-1} nor v_{i+1}. As mentioned earlier, u cannot get in conflict with v_i due to its degree, so the resulting labelling of G is good.

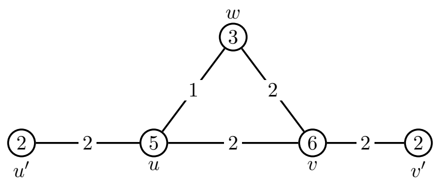
Assume now that T_{v_i} has no edge for every $i \in \{2, \dots, n-2\}$, but T_{v_1} has an edge v_1u (the case where $T_{v_{n-1}}$ has an edge is symmetrical). This means that each of $v_2, \dots, v_{n-2}$ has degree 2. In this case, we consider G' the cactus obtained from G by removing u and v_2. Note that G' has more than one edge since r has degree at least 3 in G. Then, G' is nice. By the inductive hypothesis, there is a good labelling of G'. To extend it to one of G, we must label the edges v_1u, v_1v_2, v_2v_3 so that no colour conflicts arise, and label 3 is assigned at most once. Similarly as in the previous case, this can be achieved by first labelling v_2v_3 with 1 or 2 so that no conflict between v_3 and v_4 arises, then labelling v_1v_2 with 1 or 2 so that no conflict between v_2 and v_3 arises, and lastly labelling v_1u with 1, 2 or 3 so that v_1 is not in conflict with v_0 nor v_2. $\diamond$

Due to the previous claims, in G we can assume that C is either a cycle of any length at least 3 (*i.e.*, all inner vertices have degree 2), or a triangle where one or two of its inner vertices have a pending edge attached (*i.e.*, one or two of the T_v's have size 1). We call the first of these two triangle configurations a 1-*triangle*, while we call the second configuration a 2-*triangle*. For convenience, we also regard these configurations as end-cycles, though they are technically not cycles in G.

We are now ready to conclude the proof. If G has only one cycle, then, by the previous claims and our original assumption that G is not just a cycle, it must be that G is a triangle (u, v, w, u) with a pending vertex attached to u and possibly one attached to v, in which case the claim can be verified easily (Figure 5.6 illustrates proper 2-labellings of G for these two cases). So G has at least two cycles. From now on, let us consider two cycles C_x and C_1 of G such that the block vertices b_{C_x} and b_{C_1} are two leaves at maximum distance d in $B(G^-)$. Note that C_1 is an end-cycle in G, and let r denote its root. Observe that there might be other (end-)cycles of G at distance d (in $B(G^-)$) from C_x, with root r. In case these cycles exist, we denote them by $C_2, \dots, C_q$. Then $C_1, \dots, C_q$ are end-cycles in G with the same root r, and, by how these C_i's were chosen, r either has only one neighbour u or only two neighbours u, u' of degree at least 2 that does/do not belong to the C_i's. More precisely, r is connected to the rest of the graph either via a path (through an edge ru), or via a unique cycle (containing both u and u'). Furthermore, there might be vertices



(a) G is a 1-triangle



(b) G is a 2-triangle

Figure 5.6 – Proper 2-labellings for the two cases in the proof of Theorem 5.3.15, where the cactus G is one cycle $C = (u, v, w, u)$ with one or two of its inner vertices having a pending edge attached.

of degree 1 adjacent to r. Indeed, by Claim 5.3.16, if there is a pending tree T_r attached at r, then T_r must be a star with center r. Recall that each of the C_i's is a cycle, a 1-triangle, or a 2-triangle, due to previous claims.

Now, let G' be the cactus obtained from G by removing all the non-root vertices of the C_i's (*i.e.*, all their inner vertices, plus the at most two pending vertices of the 1-triangles and 2-triangles). Since G' contains at least one cycle, it is nice, and thus, admits a good labelling by the inductive hypothesis. Our goal is to extend it to one of G by labelling the removed edges so that no conflict arises and at most a third of these edges are assigned label 3.

— Assume $q \geq 2$. We first label the edges of every C_i that is a cycle, assigning consecutive labels $2, 1, 1, 2, 2, 1, 1, \dots$ while going around, starting and ending with an edge incident to r. Note that this avoids any conflict between the inner vertices of C_i, that their colours are at most 4, and that this increases the colour of r by at least 3. For every C_i that is a 1-triangle, we assign label 2 to its two edges incident to r, and label 1 to its two other edges. Note that this raises no conflict between the inner vertices of C_i, that their colours are at most 4, and that the colour of r is increased by 4. Finally, for every C_i that is a 2-triangle, we assign label 2 to its two edges incident to r and to one pending edge, and label 1 to the two other edges. As a result, no conflict arises between inner vertices, their colours are at most 5, and the colour of r is increased by 4.
Since $q \geq 2$ and r has at least one neighbour not in the C_i's, the colour of r is at least 7, and thus, r cannot be in conflict with its neighbours in the C_i's. However, we still have to make sure that the colour of r is different from that of u (where u is the sole neighbour of r that does not belong to any of the C_i's) or from that of u and u' (where u and u' are the two neighbours of r that do not belong to any of the C_i's). Note that, in each C_i, there is an edge labelled 2 incident to r that can be relabelled 3 without causing conflicts between the inner vertices. Indeed, if C_i is a cycle, then the very first labelled edge is such an edge. If C_i is a 1-triangle or 2-triangle, then the one of its two edges labelled 2 incident to r going to the inner vertex with the largest colour, is such an edge. Thus, by changing the label from 2 to 3, of one or two of these edges, we can increment the colour of r by 1 or 2 to avoid the colours of u and u' (if it exists). This means that, by introducing at most two 3s, we can get a proper 3-labelling of G, which is good since $q \geq 2$.

— Assume $q = 1$. Assume first that C_1 is a 1-triangle or a 2-triangle. Let (r, v_1, v_2, r) denote the vertices of the cycle of C_1, and u_1 and u_2 (if it exists) denote the pending vertices attached to v_1 and v_2, respectively. We first label rv_1 and rv_2 with 1 or 2 so that no conflict arises between r and its neighbours u and u' (if it exists). This is possible since there are

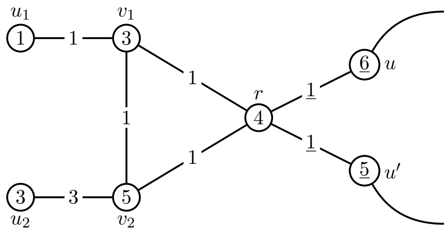


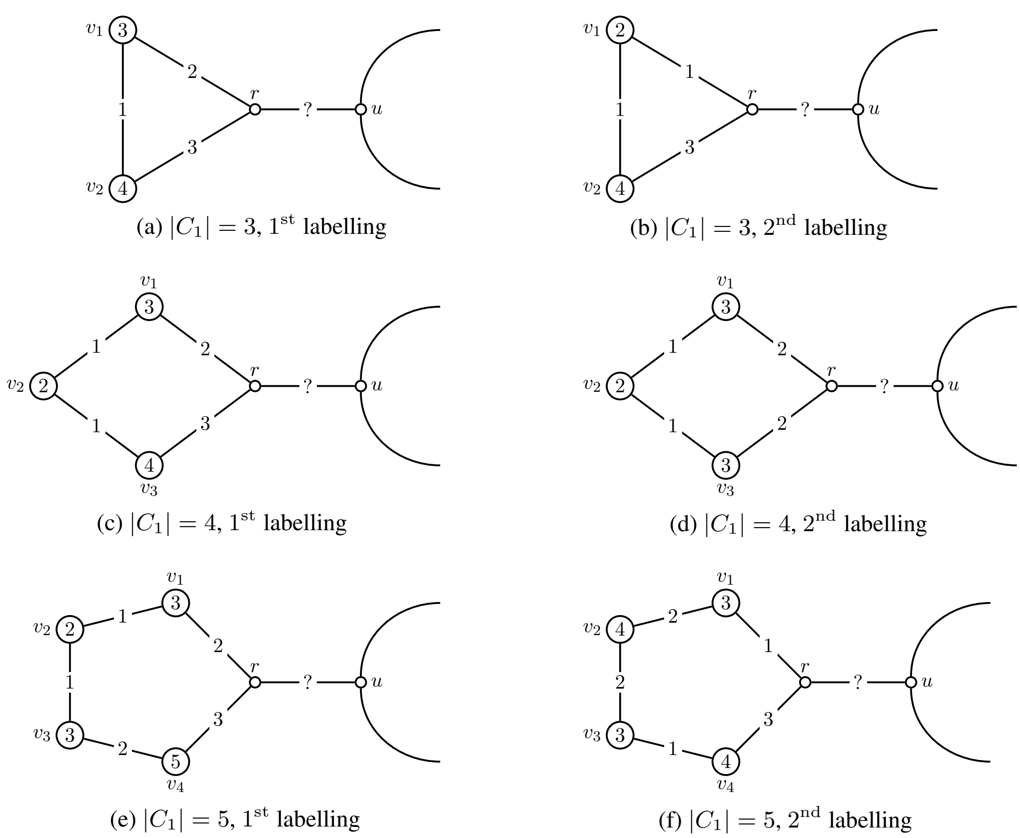
Figure 5.7 – A good labelling as described in the proof of Theorem 5.3.15 in the case where $q = 1$, C_1 is a 2-triangle, and, apart from v_1 and v_2, the vertex r has two neighbours u and u'. The underlined labels and colours correspond to the labelling provided from the inductive hypothesis (and thus, must not be modified). Now, assuming that the underlined labels and colours are as shown in the figure, we must have $c_\ell(r) = 4$, as otherwise r would be in colour conflict with either u or u'. Thus, it must be that $\ell(rv_1) = \ell(rv_2) = 1$. Then, we get that $\ell(v_1u_1) = 1$ and $\ell(v_2u_2) = 3$.

three possible combinations. In the case where C_1 is a 1-triangle, then we label v_1v_2 with 1 or 2 so that no conflict arises between v_2 and r. In the case where C_1 is a 2-triangle, then we label v_1v_2 with 1. Now, if C_1 is a 1-triangle, then we label v_1u_1 with 1, 2 or 3 so that no conflict arises between v_1 and r nor between v_1 and v_2. If C_1 is a 2-triangle, then we label v_1u_1 with 1 or 2 so that no conflict arises between v_1 and r, and then we label v_2u_2 with 1, 2 or 3 so that no conflict arises between v_2 and r nor between v_1 and v_2. In all cases, we assign label 3 to at most one edge, so the resulting proper 3-labelling of G is good since no conflict arises. Figure 5.7 illustrates a possible good labelling for the case where C_1 is a 2-triangle and both u and u' exist.

Assume now that C_1 is a cycle. First, assume that u' exists. We consider the edges of C_1, and assign to them labels 1 and 2 as previously, *i.e.*, by applying the labelling pattern $2, 1, 1, 2, 2, 1, 1, \dots$ from one edge incident to r to the other. We consider two cases:

— Assume first that, in the labelling of C_1, the two edges incident to r get assigned distinct labels (1 and 2). As earlier, no two inner vertices of C_1 are in conflict, their colours are at most 4, and, since u' exists, the colour of r is at least 5. If this raises no conflict between r and its neighbours u and u', then we are done. Otherwise, note that turning the label assigned to any of the two edges of C_1 incident to r into a 3, raises no conflict between two vertices of C_1. Since these two edges are labelled differently, one with label 1 and the other with label 2, this means that by introducing label 3 once in C_1, we can increment the colour of r by 1 or 2 so that we avoid any conflict between r and its neighbours u and u'. Then, we can deduce a good labelling of G.

— Assume now that both edges incident to r in C_1 get assigned label 2. Then, this time, the colour of r is at least 6. If there is no conflict between r and one of its neighbours u and u', then we are done. So, we can assume there is a conflict, and also that changing the label of one of the two edges of C_1 incident to r to 3, makes r in conflict with the second one of these two vertices. Then, note that we get a good labelling when labelling C_1 following the pattern $1, 2, 2, 1, 1, 2, 2, \dots$ instead, since r gets its two incident edges in C_1 being assigned label 1, the colour of r is at least 4 and smaller than the previous colours we have produced for r, and the colours of the two neighbours of r in C_1 are at most 3.



(a) $|C_1| = 3$, 1st labelling

(b) $|C_1| = 3$, 2nd labelling

(c) $|C_1| = 4$, 1st labelling

(d) $|C_1| = 4$, 2nd labelling

(e) $|C_1| = 5$, 1st labelling

(f) $|C_1| = 5$, 2nd labelling

Figure 5.8 – Labelling a pending cycle in the proof of Theorem 5.3.15. Some colours by the labelling are indicated by integers within the vertices.

Now assume u' does not exist. We start by considering the cases where C_1 has length at least 6. Start by applying the labelling pattern $2, 1, 1, 2, 2, 1, 1, \dots$ to the edges of C_1 as before. Assume first that the two edges of C_1 incident to r get assigned distinct labels. Then, change the 1 assigned as a label to one of these two edges into a 3. As a result, no conflicts arise between inner vertices of C_1, their colours are at most 5, while the colour of r is at least 6 due to the edge ru. So, the only possible conflict is between r and u. Suppose it occurs. Then, no conflict remains when assigning label 3 to the second edge of C_1 incident to r and we get a good labelling (in particular, only two edges of C_1 get assigned label 3 while its length is at least 6, and this assumption also guarantees that no two inner vertices of C_1 get in conflict). Lastly, assume that both edges of C_1 incident to r get assigned label 2 by the initial labelling scheme. Then, the colour of r is at least 5, which thus cannot be in conflict with its neighbours in C_1. If r is not in conflict with u, then we get a good labelling of G. Otherwise, we get one by changing the label of one of the two edges of C_1 incident to r to 3.

All that remains to be checked are three length values for C_1. The labelling schemes described below are illustrated in Figure 5.8.

— If C_1 has length 3 (see Figures 5.8(a) and (b)), then assigning either labels $2, 1, 3$ or $1, 1, 3$ to the edges while going around, starting and ending with r, yields a good label-

Family $\mathcal{F}$	$\exists$ **arbitrarily large** $G = (V, E) \in \mathcal{F}$: $mT(G) \geq$	$\forall G = (V, E) \in \mathcal{F}$: $mT(G) \leq$
Outerplanar 1-connected	$\frac{1}{10}\lvert E\rvert$	?
Outerplanar 2-connected	?	$\frac{1}{3}\lvert E\rvert$
Halin	?	$\frac{1}{3}\lvert E\rvert$

Table 5.2 – Summary of the results presented in Section 5.4.

ling, since r gets colour at least 6 or 5, respectively, while the inner vertices of C_1 get colours at most 4, and the colour of u is the only other colour to avoid. In particular, note that these two labelling schemes increase the colour of r in two different ways (by 5 and 4, respectively).

— If C_1 has length 4 (see Figures 5.8(c) and (d)), then we get the same conclusion from applying the labelling scheme $2, 1, 1, 3$ or $2, 1, 1, 2$. Indeed, the inner vertices get colours at most 4 and 3, respectively, while r gets colour at least 6 and 5, respectively. Also, these two schemes increase the colour of r differently, by 5 and 4, respectively.

— If C_1 has length 5 (see Figures 5.8(e) and (f)), then the sequence $2, 1, 1, 2, 3$ or $1, 2, 2, 1, 3$ yields the same conclusion. Indeed, the inner vertices get colours at most 5 and 4, respectively, while r gets colour at least 6 and 5, respectively. Also, these two schemes increase the colour of r differently, by 5 and 4, respectively.

In all cases, we can deduce a good labelling of G, which concludes the proof. □

5.4 Bounds for other graph classes

In this section, we state, in the same spirit as in the previous subsections, some lower or upper bounds on ρ_3 that can be obtained for other classes of graphs that are 3-chromatic. Indeed, we focus on outerplanar graphs and Halin graphs. Note that, strictly speaking, Halin graphs are 4-colourable, but the main part of our proof will treat the 3-chromatic ones (see upcoming Section 5.4.2 for more details). The difference between this section and the previous one, is that for the considered graph classes in this section, one of the two bounds (either the upper or the lower) is partially missing. Our results in this section are summarised in Table 5.2.

5.4.1 Outerplanar graphs

Recall that a graph is *outerplanar* if it admits a planar embedding where all vertices lie on the outer face. First off, we can obtain a result similar to Theorem 5.3.4 for outerplanar graphs.

Theorem 5.4.1. *There exist arbitrarily large connected outerplanar graphs G with $\rho_3(G) \geq 1/10$.*

Proof. For a $p \geq 1$, we construct a connected outerplanar graph G with the same properties as in the proof of Theorem 5.3.4. One possible construction (using A_1) is as follows. To obtain G, start from a star with p edges, and A_1-attach all of its p edges. Again, G has the claimed labelling properties due to Lemma 5.3.1 and Lemma 5.3.3. Also, note that G is clearly outerplanar, since the same holds true for every star, as well as A_1. □

Recall as well that outerplanar graphs form a subclass of series-parallel graphs. Thus, Theorem 5.4.1 also holds for arbitrarily large connected series-parallel graphs.

Note however that the outerplanar graphs constructed above have cut vertices. So the question remains, whether or not this lower bound still holds when considering 2-connected outerplanar graphs (recall that outerplanar graphs are 2-degenerate, and thus, each of them is either separable or 2-connected). As for an upper bound, we can prove the following:

Theorem 5.4.2. *If G is a 2-connected outerplanar graph such that $|E(G)| \geq |V(G)| + 3$, then $\rho_3(G) \leq 1/3$.*

Proof. We can assume that G is not bipartite, as otherwise the claim follows from Theorem 5.1.1. Then, $\chi(G) = 3$ since outerplanar graphs are 2-degenerate. Now, if $|V(G)|$ is odd, then the result follows from Corollary 5.2.9. So, in what follows, we assume that $|V(G)|$ is even.

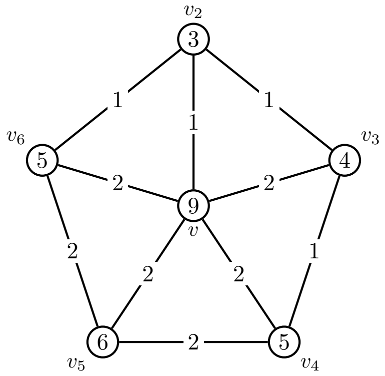
In 2-connected outerplanar graphs, the outer face forms a Hamiltonian cycle $(v_0, \dots, v_{n-1}, v_0)$. The other edges, which do not lie on the outer face, are called *chords*. Since G is not bipartite, it has an odd-length cycle C_x. Since $|V(G)|$ is even, this C_x is not the whole outer cycle of G. Furthermore, we can assume that C_x consists of consecutive vertices of the outer face, *i.e.*, that $C_x = (v_a, v_{a+1}, \dots, v_{a+x-1}, v_a)$ for some $a \in \{0, \dots, n-1\}$ (where, here and in what follows, indices are taken modulo n), or, in other words, that $v_a v_{a+x-1}$ is the only chord of G in C_x. Indeed, assume C_x has at least two chords, one of which is $v_i v_j$, where $i < j$. Note that $\{v_i, v_j\}$ is a cut set of G. This means that $V(C_x)$ is fully included in either $\{v_j, v_{j+1}, \dots, v_i\}$ or $\{v_i, v_{i+1}, \dots, v_j\}$. Assume that $V(C_x) \subseteq \{v_j, v_{j+1}, \dots, v_i\}$ (the other case being symmetrical). Then, note that $|\{v_i, v_{i+1}, \dots, v_j\}|$ must be even, as otherwise $(v_i, v_{i+1}, \dots, v_j, v_i)$ would be an odd-length cycle as desired. Now, we note that replacing $v_i v_j$ in C_x by the path $(v_i, v_{i+1}, \dots, v_j)$ results in another odd-length cycle of G with one less chord. Repeating this process as long as the resulting odd-length cycle has more than one chord, eventually we end up with an odd-length cycle of G with only one chord, which is as desired.

Up to relabelling the vertices, we can assume, w.l.o.g., that $C_x = (v_1, \dots, v_x, v_1)$. Let us consider H, the subgraph of G containing the x edges of C_x, and all the (other) edges of the Hamiltonian cycle $(v_0, \dots, v_{n-1}, v_0)$ on the outer face of G except for the edge $v_0 v_1$. Note that H is an odd unicyclic spanning subgraph of G. Since H is spanning, connected, and unicyclic, $|E(H)| = |V(G)|$, which is at most $|E(G)| - 3$, since $|E(G)| \geq |V(G)| + 3$.

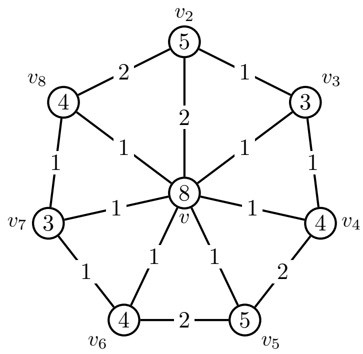
All conditions are now met to invoke the arguments in the proof of Theorem 5.2.6, from which we can deduce a proper $\{0, 1, 2\}$-labelling ℓ of G where adjacent vertices get distinct colours modulo 3, and in which only the edges of (our) H are possibly assigned label 0. Let us now consider the subgraph H' of G obtained from H by adding the edge $v_0 v_1$, which is present in G. Recall that $\ell(v_0 v_1) = 2$ by default. Note that H' contains at least two disjoint perfect matchings M_1, M_2. Indeed, since $|V(G)|$ is even, a first perfect matching M_1 of H' contains $v_0 v_1, v_2 v_3, \dots, v_{n-2} v_{n-1}$. A second perfect matching M_2 of H' contains $v_1 v_2, v_3 v_4, \dots, v_{n-1} v_0$. By Lemma 5.2.8, we can assume that at most a third of the edges in $M_1 \cup M_2$ are assigned label 0 by ℓ. Since $|M_1| + |M_2| = |E(H')| - 1 = |E(H)|$, but the edge $v_1 v_x \in E(H)$ is not included in M_1 nor M_2 (and so may have label 0 too), this gives $\mathrm{nb}_\ell(0) \leq \frac{E(H)}{3} + 1$, which is less than $|E(G)|/3$ since $|E(G)| \geq |V(G)| + 3$. More formally,

$$\mathrm{nb}_\ell(0) \leq \frac{|E(H)|}{3} + 1 = \frac{|V(G)|}{3} + 1 \leq \frac{|E(G)| - 3}{3} + 1 = \frac{|E(G)|}{3}.$$

By turning 0s by ℓ into 3s, we get a proper 3-labelling of G with the same upper bound on the number of assigned 3s. $\square$



(a) A wheel of order $n = 6 \equiv 2 \bmod 4$



(b) A wheel of order $n = 8 \equiv 0 \bmod 4$

Figure 5.9 – The proper 2-labelling for wheels of even order described in the proof of Theorem 5.4.3.

Theorem 5.4.2 does not cover all 2-connected outerplanar graphs. However, it covers all such graphs with at least three chords. Thus, to get a generalisation of Theorem 5.4.2 for all 2-connected outerplanar graphs, one has to prove a similar result for the 3-chromatic ones with at most two chords. Those with no chords are exactly cycles, for which the claim holds (see, *e.g.*, Section 4.1.6). For those with one or two chords, the claim can also be verified, for instance through considering all of the possible ways for the (at most two) chords to interact in such a 2-connected outerplanar graph, and, for each possible configuration, extending a proper 3-labelling from face to face. Let us mention that the number of cases to consider can be reduced drastically by applying some of the arguments used in the proof of Theorem 5.3.12 to deal with long threads. We voluntarily omit such a tedious proof, which, in our opinion, seems rather plausible, but would be less interesting than that of Theorem 5.4.2.

5.4.2 Halin graphs

We now proceed by proving Conjecture 2.1.9 for a 4-colourable family of graphs. A *Halin graph* is a planar graph with minimum degree 3 obtained as follows. Start from a tree T with no vertex of degree 2, and consider a planar embedding of T. Finally, add edges to form a cycle going through all the leaves of T in the clockwise order w.r.t. this embedding. A Halin graph is called a *wheel* if it is constructed from a tree T with diameter 2 (*i.e.*, T is a star).

Halin graphs are known to have many properties of interest, such as having triangles, being Hamiltonian, and having Hamiltonian cycles going through any given edge (see, *e.g.*, [119]). Also, Halin graphs are 3-degenerate, so, due to the presence of triangles, each of them has chromatic number 3 or 4. The dichotomy is well-understood, as a Halin graph has chromatic number 4 if and only if it is a wheel of even order [120]. This allows us to use our tools from Section 5.2 to establish an upper bound on ρ_3 for most Halin graphs (the 3-chromatic ones), while we can treat the remaining ones separately.

Theorem 5.4.3. *If G is a Halin graph, then $\rho_3(G) \leq 1/3$.*

Proof. First, consider the case where G is a wheel of even order n. If $n = 4$, then $G = K_4$, and the statement holds (since it can be checked by hand that $\mathrm{mT}(K_4) = 1$ and thus $\rho_3(K_4) = 1/6$).

For $n \geq 6$, we have that $\mathrm{mT}(G) = 0$. Indeed, let v be the center of the star T, and let $v_2, \dots, v_n$ be the leaves of T. We can construct a proper 2-labelling ℓ of G as follows: start from v_2v_3, and, following the edges of the cycle joining the leaves of T in increasing order of their indices, assign labels $1, 1, 2, 2, 1, 1, 2 \dots$, until v_nv_2 is labelled. If $\ell(v_nv_2) = 1$, then set $\ell(vv_2) = 1$ and $\ell(vv_i) = 2$ for every $3 \leq i \leq n$. Otherwise, if $\ell(v_nv_2) = 2$ (and so, $\ell(v_{n-1}v_n) = 1$), set $\ell(vv_2) = 2$ and $\ell(vv_i) = 1$ for every $3 \leq i \leq n$ (see Figure 5.9 for an illustration of the described labelling). It is easy to check that in both cases ℓ is a proper 2-labelling of G. Thus, $\rho_3(G) = 0$ and the statement holds.

Next, consider the case where G is not a wheel of even order. Then, $\chi(G) = 3$. If $|V(G)|$ is odd, then the result follows from Corollary 5.2.9. Thus, we can assume that $|V(G)|$ is even.

By considering any non-leaf vertex r of T in G, and defining a usual root-to-leaf (virtual) orientation, since no vertex has degree 2 in T, it can be seen that G has a triangle (u, v, w, u), where v, w are leaves in T with parent u. Furthermore, $d_G(v) = d_G(w) = 3$, while $d_G(u) \geq 3$. Due to these degree properties, note that if we consider C a Hamiltonian cycle traversing uv, then C must also include either wu or vw. More precisely, if we orient the edges of C, resulting in a spanning oriented cycle $\vec{C}$, then, at some point, $\vec{C}$ enters (u, v, w, u) through one of its vertices, goes through another vertex of the triangle, and then through the third one, before leaving the triangle. In other words, C traverses all the vertices of (u, v, w, u) at once.

Up to relabelling the vertices of (u, v, w, u), we can assume that $\vec{C}$ enters the triangle through u, then goes to v, before going to w and leaving the triangle. Let us consider H, the subgraph of G containing the three edges of (u, v, w, u), and all successive edges traversed by C after leaving the triangle except for the edge going back to u. Note that H is an odd unicyclic spanning subgraph of G, in which the only cycle is the triangle (u, v, w, u) . Furthermore, in $E(G) \setminus E(H)$, if we set $x = x_{n-3}$, then the edge xu exists. Since H is spanning, connected, and unicyclic, $|E(H)| = |V(G)|$, which is at most $2|E(G)|/3$, since $\delta(G) \geq 3$.

All conditions are now met to invoke the arguments in the proof of Theorem 5.2.6, from which we can deduce a proper $\{0, 1, 2\}$-labelling ℓ of G where adjacent vertices get distinct colours modulo 3, and in which only the edges of the chosen H are possibly assigned label 0. Let us now consider the subgraph H' of G obtained from H by adding the edge xu, which is present in G. Recall that $\ell(xu) = 2$ by default. Note that H' contains at least two disjoint perfect matchings M_1, M_2. Indeed, since $|V(G)|$ is even, then, in H, the hanging path attached at w has odd length. A first perfect matching M_1 of H' contains $x_{n-3}x_{n-4}, x_{n-5}x_{n-6}, \dots, wx_1$, and uv. A second perfect matching M_2 of H' contains $x_{n-4}x_{n-5}, x_{n-6}x_{n-7}, \dots, x_2x_1$, and wv and xu. Now, by Lemma 5.2.8, we can assume that at most a third of the edges in $M_1 \cup M_2$ are assigned label 0 by ℓ. Since $|M_1| + |M_2| = |E(H')| - 1 = |E(H)|$, this gives $\mathrm{nb}_\ell(0) \leq \frac{E(H)}{3} + 1$, which is at most $|E(G)|/3$ since $|E(G)| \geq 3|V(G)|/2$ and $|V(G)| \geq 6$ (any Halin graph has at least 4 vertices, and the only Halin graph with exactly 4 vertices is K_4, which we have already treated separately). That is,

$$\mathrm{nb}_\ell(0) \leq \frac{|E(H)|}{3} + 1 = \frac{|V(G)|}{3} + 1 \leq \frac{1}{3} \cdot \frac{2|E(G)|}{3} + 1 \leq \frac{|E(G)|}{3}.$$

By turning 0s by ℓ into 3s, we get a proper 3-labelling of G with the same upper bound on the number of assigned 3s. $\square$

Let us close this section by discussing about some of the upper bound for the value of ρ_3 for Halin graphs. We were actually not able to come up with examples of arbitrarily large Halin graphs needing many 3s in their proper 3-labellings. In fact, we are aware of only three Halin graphs that do not admit proper 2-labellings. Two of them are K_4 and the prism graph (Cartesian product of K_3 and K_2). The third one is constructed as follows: start with two perfect binary trees on 7 vertices each and add an edge between the roots (degree-2 vertices) of these trees; from the resulting tree T, construct G as explained in Section 5.4.2. All three of these graphs turn out to lie in $\mathcal{G}_1$. Thus, though we were not able to prove it, it is possible that there exists a $p \geq 1$ such that Halin graphs are in $\mathcal{G}_{\leq p}$, and even that $p = 1$.

5.5 Conclusion

This chapter was dedicated to the study of the importance of 3s in designing proper 3-labellings, this aspect being motivated by a presumption from previous works that proper 3-labellings of graphs, in general, should require only a few 3s. This led us to the introduction of the two quantifying parameters mT and ρ_3. As a main contribution, we have introduced, in Section 5.2, some tools for deducing bounds on these parameters. Applications of these, in Section 5.3, led us to results for specific classes of 3-chromatic graphs. In particular, we have established that, for several simple classes $\mathcal{F}$ of graphs, there is no $p \geq 0$ such that $\mathcal{F} \subset \mathcal{G}_{\leq p}$. In such cases, we have provided bounds on $\rho_3(\mathcal{F})$.

Several directions for further research sound particularly appealing. A first one is to prove Conjecture 2.1.9 for more classes of graphs, or to exhibit better upper bounds towards it. Another one is to investigate whether the bound of $1/3$ in that conjecture is close to being tight or not, in general. Indeed, at the moment we only know of two small connected graphs, namely C_3 and C_6, which attain the bound, while the class of arbitrarily large graphs with the biggest value ρ_3 we could construct, achieves a ratio of $1/10$ (Theorem 5.3.4).

An interesting perspective could be to provide better lower bounds, *i.e.*, find graphs requiring even more 3s in their proper 3-labellings. This could be done through using Lemma 5.3.3 (just as in Theorem 5.3.4 for instance) with graphs H that are better than those used throughout this chapter. In particular, it would be interesting to find such graphs H with similar properties to A_1 and A_2, but with $\rho_3(H) > 1/10$. Other properties of interest for H include large density. Note that the graphs we construct, for instance, in the proof of Theorem 5.3.4, are rather sparse due to how H-augmentations are performed. It is not always true, however, that performing H-augmentations results in sparse graphs. For example, consider A_2-augmenting a small number of edges of a huge complete graph. Following these thoughts, we wonder whether denser versions of A_1 and A_2 exist. Another property of interest could be high connectivity. As mentioned after the proof of Theorem 5.3.5, the graphs A_1 and A_2 can be used to produce 2-connected graphs. However, these graphs cannot be used to produce graphs with connectivity at least 3.

Chapter 6

Finding a largest locally irregular induced subgraph

In this chapter we introduce and study the problem of finding a largest locally irregular induced subgraph of a given graph G. Equivalently, given a graph G, find a subset S of $V(G)$ with minimum order, such that deleting the vertices of S from G results in a locally irregular graph; we denote with $\mathrm{I}(G)$ the order of such a set S. We first examine some simple graph families. We then show that the decision version of the introduced problem is NP-Complete, even for restricted families of graphs. Moreover, we cannot even approximate an optimal solution within a ratio of $\mathcal{O}(n^{1-\frac{1}{k}})$, for every $k \geq 1$, where n is the order the graph, unless P=NP, even when the input graph is bipartite.
For positive results, we provide two FPT algorithms for computing $\mathrm{I}(G)$, the first one considering, as a parameter, the size of the solution k and the maximum degree Δ of G, and the second one considering the treewidth tw and Δ of G, with running times $(2\Delta)^k n^{\mathcal{O}(1)}$ and $\Delta^{4tw} n^{\mathcal{O}(1)}$ respectively. We then prove that there is no algorithm that computes $\mathrm{I}(G)$ with dependence $f(k)n^{o(k)}$ or $f(tw)n^{o(tw)}$, unless the ETH fails, showing that our algorithms are essentially optimal.
This chapter presents a joint work with N. Melissinos and T. Triomatis, presented in [69].

The problem we introduce in this chapter belongs to a more general and well studied family of problems, which is about identifying a largest induced subgraph of a given graph that verifies a specific property Π. That is, given a graph $G = (V, E)$ and an integer k, is there a set $V' \subseteq V$ such that $|V'| \leq k$ and $G[V \setminus V']$ has the specified property Π? In our case, the property Π is "the induced subgraph is locally irregular". This generalised problem is indeed classic in graph theory, and it is known as the INDUCED SUBGRAPH WITH PROPERTY Π (ISPΠ for short) problem in [74]. Unfortunately, it was shown in [91], that ISPΠ is a hard problem for any property Π that is *hereditary*, *i.e.*, all induced subgraphs of G verify Π if G itself verifies that property.

However, the ISPΠ problem remains interesting even if the property Π is not hereditary (as is the case for the property of interest in this chapter). Recently, [19] studied the problem for Π being "all vertices of the induced subgraph have odd degree", which clearly is not a hereditary property. Nevertheless, they showed that this is an NP-hard problem, and they gave an FPT algorithm that solves the problem when parameterised by the rank-width. Also, [3, 11, 98] studied the ISPΠ problem, where Π is the natural property "the induced subgraph is d-regular", where d is an integer given in the input. In particular, in [11] it is shown that finding a largest (connected) induced subgraph that is d-regular, is NP-hard to approximate, even when restricted on bipartite or planar graphs. [11] also provides a linear-time algorithm to solve this problem for graphs with bounded treewidth. In contrast, [3] takes a more practical approach, as they focus on solving the problem for the particular values of $d = 1$ and $d = 2$, by using bounds from quadratic programming, Lagrangian relaxation and integer programming.

It is quite clear that, in some sense, the property that interests us lies on the opposite side of the one studied in [3, 11, 98]. However, both properties, "the induced subgraph is regular" and "the induced subgraph is locally irregular" are not hereditary. This means that we do not get an NP-hardness result directly from [91]. Furthermore, the ISPΠ problem always admits an FPT algorithm if Π is a hereditary property [44, 83], but for a non-hereditary one, this is not always true. Indeed in [98], the authors proved that when considering Π as "the induced subgraph is regular", the ISPΠ problem is W[1]-hard when parameterised by the size of the solution. That is, there should be no $\mathcal{O}^*(f(k)n^c)$ time algorithm for this problem, where c is a constant. For such problems, it is also interesting to see if there exists any algorithm with running time $\mathcal{O}^*(n^{o(k)})$ or $f(k)n^{o(k)}$. The authors of [51, 52, 53] provide techniques that can be used to strongly indicate the non-existence of such algorithms, by applying them on a variety of W[1]-hard and W[2]-hard problems, such as the INDEPENDENT SET and the DOMINATING SET, parameterised by the size of their solutions. These lower bounds are shown under the assumption of the EXPONENTIAL TIME HYPOTHESIS. Intuitively, this hypothesis claims that there can be no algorithm for solving 3-SAT that does not exhaustively search through an exponential number of possible solutions. A weaker, but widely utilised, version of this hypothesis, claims that SAT cannot be solved in time $2^{o(n+m)}$.

We begin in Section 6.1 by providing the basic notations and definitions that are going to be used throughout this chapter. In Section 6.2, we deal with the complexity of the introduced problem. In particular, we show that the problem belongs to P if the input graph is a path, cycle, complete bipartite or complete graph. We then prove that finding a largest induced locally irregular subgraph of a given graph G is NP-hard, even if G is restricted to being a subcubic planar bipartite, or a cubic graph.

As the problem we introduce seems to be computationally hard even for restricted families of graphs, we proceed by investigating its approximability. Unfortunately, we prove in Section 6.3 that for any bipartite graph G of order n and $k \geq 1$, there can be no polynomial-time algorithm that finds an approximation of $\mathrm{I}(G)$ within ratio $\mathcal{O}(n^{1-\frac{1}{k}})$, unless P=NP. Nevertheless, we do manage to give a (simple) d-approximation algorithm for d-regular bipartite graphs.

We then look into parameterised complexity. In Section 6.4, we present two algorithms that compute $\mathrm{I}(G)$, each one considering different parameters. The first considers the size of the solution k and the maximum degree Δ of G, and has running time $(2\Delta)^k n^{\mathcal{O}(1)}$, while the second considers the treewidth tw and Δ of G, and has running time $\Delta^{4tw} n^{\mathcal{O}(1)}$. Unfortunately, these algorithms can be considered as being FPT only if Δ is part of the parameter. In Section 6.4.2, we present two linear fpt-reductions which prove that the problem is W[2]-hard when parameterised only by the size of the solution and W[1]-hard when parameterised only by the treewidth. These reductions also show that we cannot even have an algorithm that computes $\mathrm{I}(G)$ in time $f(k)n^{o(k)}$ or $\mathcal{O}^*(f(tw)n^{o(tw)})$, unless the ETH fails.

6.1 Preliminaries

Let $G = (V, E)$ be a graph. Now, let $S \subseteq V$ be such that $G[V \setminus S]$ is a locally irregular graph; any set S that has this property is said to be an *irregulator of* G. Moreover, let $\mathrm{I}(G)$ be the minimum order that any irregulator of G can have. We will say that S is a *minimum* irregulator of G if S is an irregulator of G and $|S| = \mathrm{I}(G)$.

We also define the following, which generalises the notion of an irregulator. Let $G = (V, E)$ be a graph, $S, X \subseteq V$ and let $G' = G[V \setminus S]$. If S is such that for every two adjacent vertices u, v in $X \setminus S$, we have that $d_{G'}(u) \neq d_{G'}(v)$, then S will be called an *irregulator of* X *in* G. We define the notion of a minimum irregulator of X in G analogously to the previous paragraph. If S is a minimum irregulator of X in G, then we define $\mathrm{I}(G, X) = |S|$.

We now provide some lemmata and an observation that will be useful throughout this chapter. In the three lemmata below, we investigate the relationship between $\mathrm{I}(G)$ and $\mathrm{I}(G, X)$.

Lemma 6.1.1. *Let* $G = (V, E)$ *be a graph and let* $X \subseteq V$. *Then,* $\mathrm{I}(G, X) \leq \mathrm{I}(G)$.

Proof. Let S be a minimum irregulator of G, $G' = G[V \setminus S]$ and $X' = X \setminus S$. Observe that for every pair of vertices u, v such that $u \in X'$ and $v \in N_{G'}(u) \cap X'$, we have that $d_{G'}(u) \neq d_{G'}(v)$, since S is a minimum irregulator of G. It follows that S is also an irregulator of X in G, and thus we have that $\mathrm{I}(G, X) \leq |S| = \mathrm{I}(G)$. □

Lemma 6.1.2. *Let* $G = (V, E)$ *be a graph and* $S, X \subseteq V$ *such that* S *is a minimum irregulator of* X *in* G. *Then,* $S \subseteq N[X]$ *and* $\mathrm{I}(G, X) = \mathrm{I}(G[N[X]], X)$.

Proof. Let S be a minimum irregulator of X in G, $S_1 = S \cap N[X]$ and $S_2 = S \setminus S_1$. It suffices to prove that S_1 is an irregulator of X in G. Indeed, if $S_1 \subseteq S$ verifies this property, then, since S is a minimum irregulator of X in G, we can conclude that $S = S_1$ and that $S \subseteq N[X]$ (by definition of S_1).

Assume now that S_1 is not an irregulator of X in G. Then there exists a pair of vertices u, v where uv is an edge in $G[X \setminus S_1]$ and $d_{G[V \setminus S_1]}(u) = d_{G[V \setminus S_1]}(v)$. Observe that $N[\{u, v\}] \subseteq N[X]$, and thus $N[\{u, v\}] \cap S_2 = \emptyset$. Therefore, $d_{G[V \setminus S]}(u) = d_{G[V \setminus S_1]}(u) = d_{G[V \setminus S_1]}(v) = d_{G[V \setminus S]}(v)$. This is a contradiction since S is a minimum irregulator of X in G.

Now, we prove that $\mathrm{I}(G, X) = \mathrm{I}(G[N[X]], X)$. Let S be a minimum irregulator of X in G. Since $S \subseteq N[X]$ and any vertex $v \in X \setminus S$ has $N(v) \subseteq N[X]$, we have that $d_{G[V \setminus S]}(v) = d_{G[N[X] \setminus S]}(v)$. Thus, S is an irregulator of X in $G[N[X]]$ and $\mathrm{I}(G, X) \geq \mathrm{I}(G[N[X]], X)$. Now for the opposite direction, let S' be a minimum irregulator of X in $G[N[X]]$. We now show that S' is also an irregulator of X in G. This follows from the fact that for all $v \in X \setminus S'$, we have $d_{G[V\setminus S']}(v) = d_{G[N[X]\setminus S']}(v)$ (again because $N(v) \subseteq N[X]$). Therefore, $\mathrm{I}(G,X) \leq \mathrm{I}(G[N[X]],X)$. □

Lemma 6.1.3. *Let $G = (V,E)$ be a graph, and $X_1,\dots,X_n \subseteq V$ such that $N[X_i] \cap N[X_j] = \emptyset$ for every $1 \leq i < j \leq n$. Then $\sum_{i=1}^{n} \mathrm{I}(G,X_i) \leq \mathrm{I}(G)$.*

Proof. Let $X = \bigcup_{i=1}^{n} X_i$. For every $1 \leq i \leq n$, let S_i be a minimum irregulator of X_i in G and $G'_i = G[V \setminus S_i]$, and let $S = \bigcup_{i=1}^{n} S_i$ and $G' = G[V \setminus S]$. Observe first that for every $i \neq j$, since $N[X_i] \cap N[X_j] = \emptyset$, we have that $S_i \cap S_j = \emptyset$ as well. Thus, $|S| = \sum_{i=1}^{n} |S_i|$.

We now show that S is a minimum irregulator of X in G. Assume that there exists an S' such that $|S'| < |S|$ and S' is an irregulator of X in G. Then, there exists a $k \leq n$ such that the set $S'_k = S' \cap N[X_k]$, is such that $|S'_k| < |S_k|$, as otherwise $|S'|$ cannot be smaller than $|S|$. Observe that S'_k must be an irregulator of X_k in G; this holds because for any vertex $u \in S' \setminus S'_k$, we know that $u \notin N[X_k]$. This is a contradiction since we have assumed that S_k is a minimum irregulator of X_k in G and S'_k is an irregulator of X_k in G of order smaller than that of S_k. Therefore, S is a minimum irregulator of X in G, and the statement follows by Lemma 6.1.1. □

Lemma 6.1.4. *Let $G = (V,E)$ be a graph, X be a subset of V and S be an irregulator of G. The set $S \cap N[X]$ is an irregulator of X in G and an irregulator of X in $G[N[X]]$.*

Proof. Let $S_X = S \cap N[X]$, $G' = G[V \setminus S]$ $G^* = G[V \setminus S_X]$. Assume that S_X is not an irregulator of X in G. Then there exist two adjacent vertices v, u such that $\{v,u\} \subset X \setminus S_X$ and $d_{G^*}(u) = d_{G^*}(v)$. Since $S_X = S \cap N[X]$ we have that $N_{G[S]}(u) = N_{G[S_X]}(u)$ and $N_{G[S]}(v) = N_{G[S_X]}(v)$. Therefore $d_{G'}(u) = d_{G^*}(u) = d_{G^*}(v) = d_{G'}(u)$ which is a contradiction since G' is locally irregular. It remains to show that S_X is an irregulator of X in $G[N[X]]$. Note that for any vertex $v \in X \setminus S_X$, $N[v]$ is included in both G and $G[N[X]]$. Therefore, $d_{G^*}(v) = d_{G[N[X]\setminus S_X]}(v)$ since we have removed the same vertices from $N[X]$. The result follows. □

Observation 6.1.5. *Let $G = (V,E)$ be a graph and S be an irregulator of G. Then, for every edge $uv \in E$, if $d_G(u) = d_G(v)$, then S contains at least one vertex in $N[\{u,v\}]$. Additionally, for a set $X \subseteq V$, let S^* be an irregulator of X in $G[N[X]]$. Then, for every edge $uv \in E(G[X])$, if $d_{G[X]}(u) = d_{G[X]}(v)$, then S^* contains at least one vertex in $N[\{u,v\}]$.*

6.2 (Classical) complexity

In this section, we deal with the (classical) complexity of the problem we introduced. First, we calculate $\mathrm{I}(G)$ for some simple families of graphs. Specifically, we show that $\mathrm{I}(G)$ can be calculated in linear time when G is a path, a cycle and a complete or a complete bipartite graph. Then, we show that finding a minimum irregulator of a graph is NP-hard. This remains true even for quite restricted families of graphs, such as cubic graphs and subcubic planar bipartite graphs.

6.2.1 Polynomial cases

Theorem 6.2.1. *Let G be a graph. If $G = K_n$, then $\mathrm{I}(G) = n - 1$. Also, if $G = K_{n,m}$ with $0 \leq n \leq m$, then $\mathrm{I}(G) \leq 1$ with the equality holding if and only if $n = m$.*

Proof. Let $G = (V, E)$. Assume that $G = K_n$, and let S be an irregulator of G with $|S| < n - 1$. Then $G' = G[V \setminus S]$ is a complete graph of order $n' > n - (n-1) = 1$, and for any $n' \geq 2$, we have that $K_{n'}$ is not locally irregular, leading to a contradiction.

Observe that $K_{n,m}$, with $0 \leq n < m$, is locally irregular, and thus $\mathrm{I}(K_{n,m}) = 0$ in this case. Assume now that $G = K_{n,n}$ with $n \geq 1$. We have that $\mathrm{I}(G) \geq 1$ as $K_{n,n}$ is not locally irregular. Let L, R be the two bipartitions of V, with $|L| = n$ and $|R| = n$. Consider the set $S = \{v\}$, where v is any vertex of L. Clearly, after the deletion of v, the graph $G' = G[V \setminus S]$ is isomorphic to $K_{n-1,n}$ which is locally irregular. □

Theorem 6.2.2. *If P_n is the path on n vertices, then*

$$\mathrm{I}(P_n) = \begin{cases} \lfloor \frac{n}{4} \rfloor, & \text{if } n \not\equiv 2 \bmod 4 \\ \lfloor \frac{n}{4} \rfloor + 1, & \text{if } n \equiv 2 \bmod 4 \end{cases}$$

Proof. We begin our proof by examining the cases of P_1, P_2, P_3, and P_4. Observe first of all that P_1 and P_3 are locally irregular graphs. It follows that $\mathrm{I}(P_1) = \mathrm{I}(P_3) = 0$.

On the other hand, it is also easy to check that P_2 is not locally irregular, but that deleting any one of its vertices suffices to turn it into P_1 (which is locally irregular). It follows that $\mathrm{I}(P_2) = 1$. We now show that $\mathrm{I}(P_4) = 1$. Let $P_4 = (v_1, v_2, v_3, v_4)$ and note that P_4 is not locally irregular (we have that $d(v_2) = d(v_3) = 2$). Moreover, deleting either v_1 or v_4 from P_4, results in the graph P_3, which is locally irregular. Thus $\mathrm{I}(P_4) = 1$. Observe moreover that any path on more than 4 vertices is not locally irregular.

We are now ready to continue with the proof. Let $n, k, d \in \mathbb{N}$, with $n \geq 5$, $n \equiv k \bmod 4$, $d = \lfloor \frac{n}{4} \rfloor$ and $G = P_n = v_1 \dots v_n$. We have the following two cases:

- Case $k \neq 2$. Consider the set $S = \{v_i : i \equiv 0 \bmod 4\}$. We have that $|S| = d$. Also, observe that the graph $G[V(G) \setminus S]$ has d connected components, each one of which is isomorphic to P_3, which are locally irregular, and a connected component isomorphic to P_k, where $k \in \{0, 1, 3\}$, which is also locally irregular (the graph P_0 is the empty graph). It follows that S is an irregulator of P_n and that $\mathrm{I}(P_n) \leq |S| = d$. All that is left to show is that $\mathrm{I}(P_n) \geq d$. Let us assume that there exists a set S_0 that is an irregulator of P_n and $|S_0| < d$. Now observe that $G[V(G) \setminus S_0]$ contains at least one connected component isomorphic to P_m, with $m \geq 4$ This is a contradiction, since P_m is not locally irregular.
- Case $k = 2$. Consider the set $S = \{v_i : i \equiv 0 \bmod 4\} \cup \{v_n\}$. We have that $|S| = d + 1$. Similarly to the previous case, we have that $G[V(G) \setminus S]$ contains d connected components isomorphic to P_3 and one connected component isomorphic to P_1. Thus S is an irregulator of P_n and $\mathrm{I}(P_n) \leq |S| \leq d + 1$. All that is left to show is that $\mathrm{I}(P_n) \geq d + 1$. Observe that the arguments supporting that $\mathrm{I}(P_n) > d$ are the same as the previous case. So, we assume that there exists a set S_0 that is an irregulator of P_n and $|S_0| = d$. Observe that all the connected components of $G[V(G) \setminus S_0]$ are paths. Also, if there exists a connected component isomorphic to a P_m, with $m \geq 4$, then $G[V(G) \setminus S_0]$ is not locally irregular. So we may assume that all the connected components of $G[V(G) \setminus S_0]$ are isomorphic to a path on at most 3 vertices. It follows that one of these components must be isomorphic to P_2. This is a contradiction since P_2 is not locally irregular.

□

Corollary 6.2.3. *If C_n is the cycle on $n \geq 3$ vertices, then $\mathrm{I}(C_n) = \mathrm{I}(P_{n-1}) + 1$, where P_{n-1} is the path on $n - 1$ vertices.*

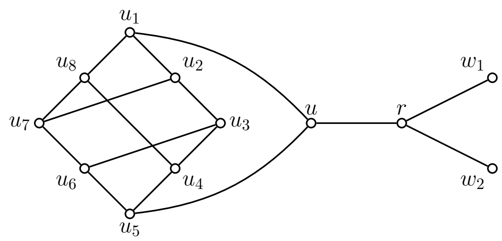


Figure 6.1 – The 3-gadget used in the proof of Theorem 6.2.4.

Proof. Observe that for every vertex v belonging to the cycle $G = C_n$, we have $d(v) = 2$. It follows from Observation 6.1.5 that $\mathrm{I}(C_n) \geq 1$ and that any S that is an irregulator of G contains at least one vertex, say vertex v. The statement follows by observing that the graph $G[V(G) \setminus v]$ is isomorphic to P_{n-1}. □

6.2.2 Hard cases

We now show that finding a minimum irregulator of a graph is NP-hard. This remains true even for quite restricted families of graphs, such as cubic (*i.e.*, 3-regular), and subcubic planar bipartite graphs, *i.e.*, planar bipartite graphs of maximum degree at most 3.

Theorem 6.2.4. *Let G be a graph and $k \in \mathbb{N}$. Deciding if $\mathrm{I}(G) \leq k$ is NP-complete, even if G is a cubic graph.*

Proof. Since the problem is clearly in NP, we focus on proving it is also NP-hard. The reduction is from 2-BALANCED 3-SAT, which was proven to be NP-complete in [36]. In that problem, a 3CNF formula F is given as an input, comprised by a set C of clauses over a set of Boolean variables X. In particular, we have that each clause contains exactly three literals, and each variable $x \in X$ appears in F exactly twice as a positive and twice as a negative literal. The question is whether there exists a truth assignment to the variables of X satisfying F.

Let F be a 3CNF formula with m clauses $C_1, \dots, C_m$ and n variables $x_1, \dots, x_n$ that is given as input to the 2-BALANCED 3-SAT problem. We construct a cubic graph G such that F is satisfiable if and only if $\mathrm{I}(G) \leq 3n$. To construct $G = (V, E)$, we start with the following graph: for each literal x_i ($\neg x_i$ resp.) in F, add a *literal vertex* v_i (v'_i resp.) in V, and for each clause C_j of F, add a *clause vertex* c_j in V. Next, for each $1 \leq j \leq m$, add the edge $v_i c_j$ ($v'_i c_j$ resp.) if the literal x_i ($\neg x_i$ resp.) appears in C_j according to F. Observe that the resulting graph is bipartite, for each clause vertex c we have $d(c) = 3$ and for each literal vertex v we have $d(v) = 2$ (since in F, each variable appears twice as a positive and twice as a negative literal). To finish the construction of G, we make use of the 3-gadgets, illustrated in Figure 6.1. When we say that we *attach* a copy H of the 3-gadget to the vertices v_i and v'_i (for some $1 \leq i \leq n$), we mean that we add H to G, and we identify the vertices w_1 and w_2 to the vertices v_i and v'_i respectively. Now, for each pair of literal vertices $\{v_i, v'_i\}$, attach one copy H_i of the 3-gadget to the vertices v_i and v'_i (see Figure 6.2). Clearly this construction is achieved in linear time in regards to $n + m$. Note also that the resulting graph G is cubic. Before we move on with the reduction, we state the following claim:

Claim 6.2.5. *Let H be a copy of the 3-gadget, shown in Figure 6.1, and $X = V(H) \setminus \{w_1, w_2\}$. We have the following:*

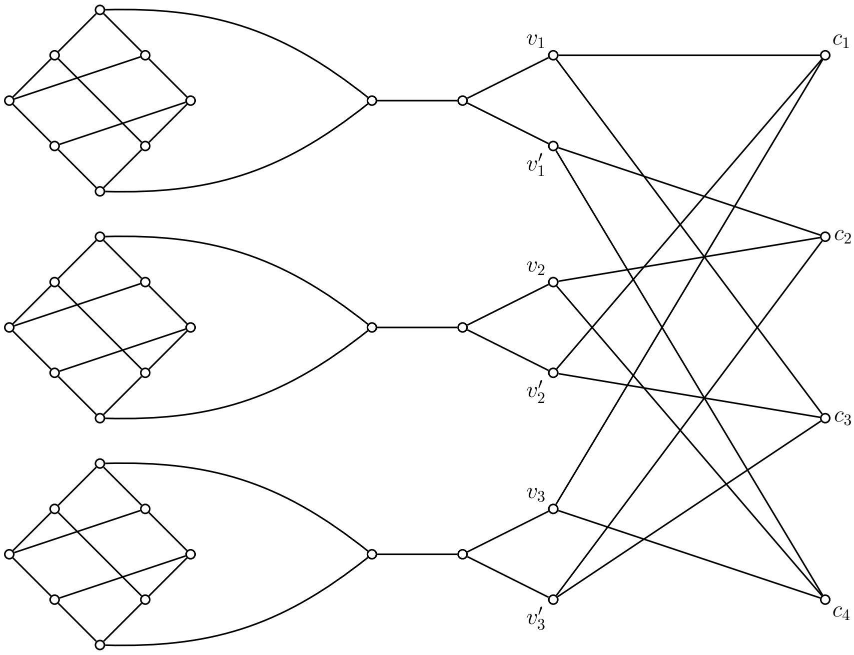


Figure 6.2 – An example of the construction of the cubic graph G in the proof of Theorem 6.2.4, starting from the input formula $F = (x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee x_2 \vee \neg x_3) \wedge (x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_1 \vee x_2 \vee x_3)$. The formula has $n = 3$ variables x_1, x_2, x_3, and $m = 4$ clauses. For $1 \leq i \leq n$, the vertex v_i (v_i' resp.) corresponds to the appearances of the literal x_i ($\neg x_i$ resp.) in F.

— $\mathrm{I}(H, X) = 3$*;*
— *for any S that is a minimum irregulator of X in H, we have that $r \notin S$;*
— *for $w \in \{w_1, w_2\}$, if $w \in S$, with S being a minimum irregulator of X in H, then $S = \{u_4, u_8, w\}$. Furthermore, these are the only optimal irregulators of X in H that contain either w_1 or w_2.*

Proof of the claim. First we show that if S is an irregulator of X in H, then $|S| \geq 3$. Clearly, if $|S| = 1$, then S cannot be an irregulator of X in H. Assume now that $|S| = 2$ and consider the edges ur, u_2u_7 and u_4u_5. These edges have in common that both of their incident vertices have the same degree (which is equal to 3). It follows from Observation 6.1.5 that S contains at least one vertex in each one of the sets $S_1 = N(uv) = \{u_1, u_5, u, r, w_1, w_2\}$, $S_2 = N(u_2u_7) = \{u_1, u_2, u_3, u_6, u_7, u_8\}$ and $S_3 = N(u_4u_5) = \{u_3, u_4, u_5, u_6, u_8\}$. Assume first that $u_1 \in S$. In Figure 6.3 we illustrate all possible subsets of vertices of H of order 2 that contain u_1. Clearly none of them is an irregulator of X in H. It follows that $u_1 \notin S$. Note that due to symmetry, we can also deduce that $u_5 \notin S$. It follows that $S \cap S_1 \subseteq \{u, r, w_1, w_2\}$ and, since $|S| = 2$, that the remaining vertex w of S belongs to $(S_2 \cap S_3) \setminus \{u_1, u_5\} = \{u_3, u_6, u_8\}$. Let $H' = H[V(H) \setminus S]$. It is easy to see that if $w = u_3$ then $d_{H'}(u_7) = d_{H'}(u_8)$, if $w = u_6$ then $d_{H'}(u_3) = d_{H'}(u_4)$ and that if $w = u_8$ then $d_{H'}(u_3) = d_{H'}(u_6)$ (and this holds true for any possible combination of vertices $w \in \{u_3, u_6, u_8\}$ and $w' \in \{u, r, w_1, w_2\}$). Thus S cannot be an irregulator of X in H and $\mathrm{I}(H, X) \geq 3$. For the remaining claims of the statement, it suffices to find all the irregulators of X in H of order 3. By doing an exhaustive search, we were able to identify these irregulators.

They are (up to symmetry) the following: $\{u_1, u_8, u_7\}$, $\{u_1, u_5, u_7\}$, $\{u_1, u_3, u_7\}$, $\{u_4, u_8, u\}$, $\{u_4, u_8, w_1\}$ and $\{u_4, u_8, w_2\}$. $\diamond$

We are now ready to show the equivalence between finding a satisfying assignment ϕ of F, and finding an S that is an irregulator of G such that $|S| = 3n$ (from which follows that $\mathrm{I}(G) \leq 3n$).

Let ϕ be a satisfying assignment of F, and let S' be the set of literal vertices v_i (v'_i resp.) such that the corresponding literals x_i ($\neg x_i$ resp.) are assigned value *true* by ϕ. Now, for each copy H_i of the gadget which was used in the construction of G, let $S'_i = \{u_4, u_8, \alpha\}$, where $\alpha \in S'$, and consider the set $S = \bigcup_{i=1}^{n} S'_i$. Note that $|S| = 3n$. We now show that S is an irregulator of G. Since ϕ is a satisfying assignment of F, each clause C_j contains at least one literal that is set to *true*. In other words, the clause vertex c_j is adjacent to at least one literal vertex that belongs to S. Let $G' = G[V(G) \setminus S]$, and note that $d_{G'}(c_j) \leq 2$ (for every $1 \leq j \leq m$), while the degree of all the literal vertices of G' is equal to 3. It follows, from the previous observations and from Claim 6.2.5, that S is an irregulator of G. Furthermore, by the construction of S, we know that for any literal vertex v of G', the copy of the vertex r which is incident to v has degree 2 in G'. Finally, observe that $S \cap V(H_i)$ is an irregulator of X_i in H_i (by the construction of S) and that deleting any vertex of G that does not belong to $N[X_i]$, does not change the degree of any vertex in $N[X_i]$. So, we can conclude that there are no two vertices in X_i that have the same degrees in G'. Thus S is an irregulator of G.

For the other direction, assume that $\mathrm{I}(G) \leq 3n$ and let S be a minimum irregulator of G. For $1 \leq i \leq n$, let $X_i = V(H_i) \setminus \{v_i, v'_i\}$ and observe that $N[X_i]$ contains exactly the vertices of the gadget H_i. Also, let $S_i = S \cap V(H_i)$, for all $i \in \{1, \dots, n\}$. First we are going to prove some properties of S:

Claim 6.2.6. *For the given set S, the following properties hold for all $i \in \{1, \dots, n\}$:*

1. *S_i is a minimum irregulator of X_i in H_i.*
2. *$S = \bigcup_{i=1}^{n} S_i$.*
3. *The vertex r belonging to the gadget H_i, does not belong to S.*
4. *If $v_i \in S$ (for some i), then $v'_i \notin S$ and* vice versa.
5. *For all $j \in \{1, \dots, m\}$, we have that $c_j \notin S$.*

Proof of the claim. For the first item, let us first show that S_i is an irregulator of X_i in H_i. Assume that S_i does not verify this property; then there exist two vertices u, v in $X_i \setminus S_i$ that have the same degree in $G[V \setminus S_i]$. Since $S \setminus S_i$ does not include any vertices of H_i, we know that u, v belong in $G[V \setminus S]$. This is a contradiction since S is a minimum irregulator of G. In order to show that S_i is actually a minimum irregulator of X_i in H_i, we need to take in consideration the order of S. Since S_i is an irregulator of X_i in H_i, we have that $|S_i| \geq 3$. Assume that there exists an i such that $|S_i| > 3$. Since $S_i \subseteq N_G[X_i]$ and $N_G[X_i] \cap N_G[X_j] = \emptyset$, for every $i \neq j$, we have that $|S| \geq \sum_{i=1}^{n} |S_i| > 3n$. This is a contradiction because $|S| \leq 3n$. Thus the first item holds.

The rest of the items follow from the first item. For the second item, we just need to observe that $\sum_{i=1}^{n} |S_i| = 3n$ and $|S| \leq 3n$. Therefore, S cannot contain any other vertex. The third and four items follow from the facts that $S_i = S \cap V(H_i)$, the first item and Claim 6.2.5. Finally, the fifth item holds because $c_j \notin N_G[X_i]$, for any $i \in \{1, \dots, n\}$ and $j \in \{1, \dots, m\}$, and $S = \bigcup_{i=1}^{n} S_i \subseteq \bigcup_{i=1}^{n} N_G[X_i]$. $\diamond$

Before we give the truth assignment let us note two more things for some vertices of $G' = G[V \setminus S]$. First, any literal vertex v that belongs to G', has $d_{G'}(v) = 3$ since S does not contain

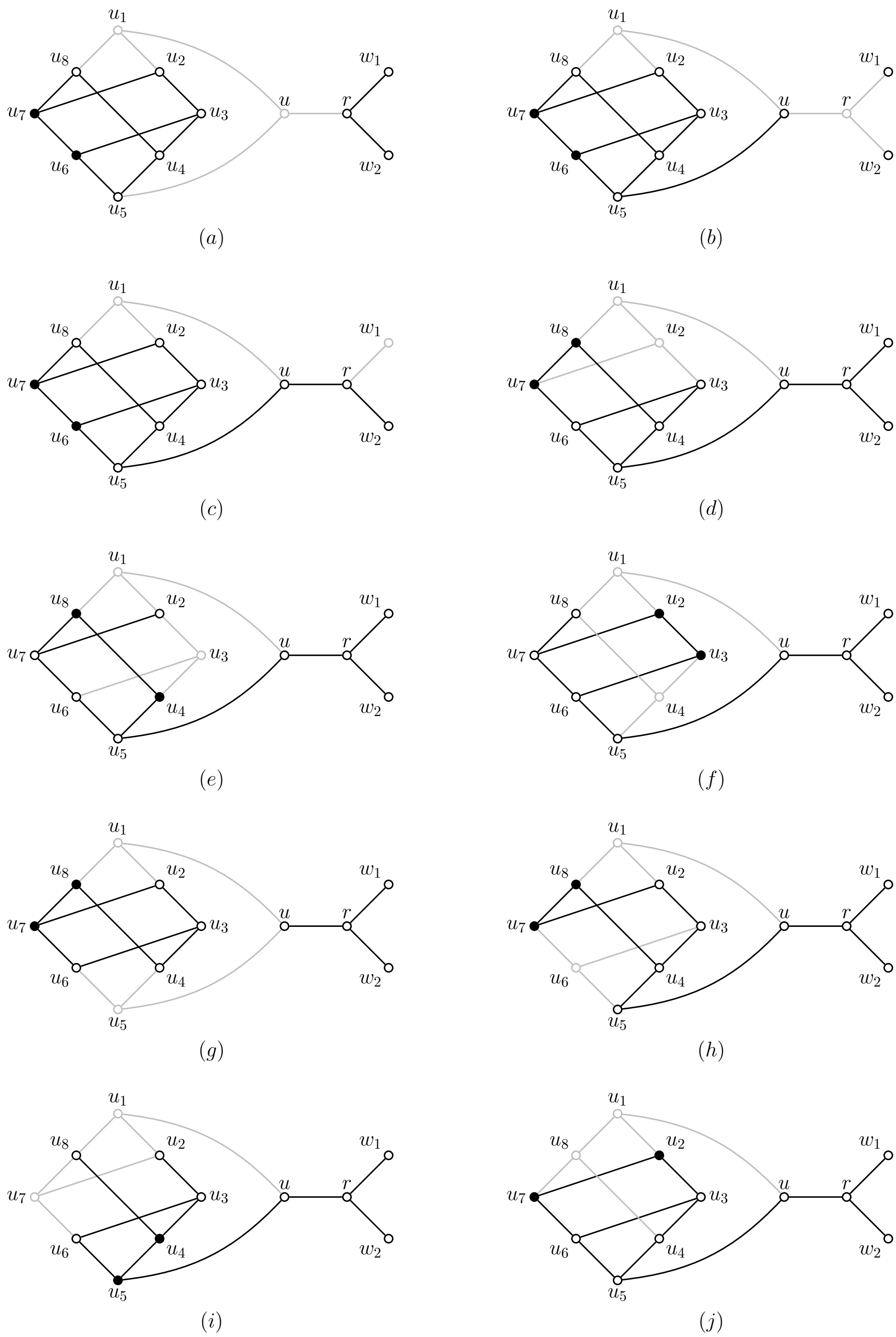


Figure 6.3 – An illustration of all the possible $S \subseteq V(H)$ of order 2 that contain u_1, used in the proof of Claim 6.2.5. The case $S = \{u_1, w_2\}$ is omitted because it is the same as case (c). Gray vertices and edges represent the vertices and edges that do not appear in $H[V(H) \setminus S]$. Any pair of black vertices have the same degree in $H[V(H) \setminus S]$.

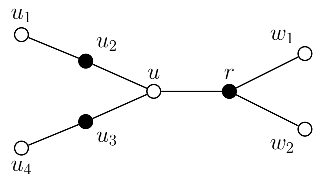


Figure 6.4 – The gadget used in the proof of Theorem 6.2.7. The white and black vertices are used to denote vertices belonging to different classes of the bipartition.

any of the neighbours of v. Furthermore, for each clause vertex c_j, there must exist a literal vertex $v \in N(c_j)$ such that $v \in S$, as otherwise c_j would have the same degree as all its neighbours in G'.

Now consider the following truth assignment: we assign the value *true* to every variable x_i if the corresponding literal vertex v_i belongs to S, and value *false* to every other variable. Now, since for every $1 \leq j \leq m$ we have that $d_{G'}(c_j) < 3$, it follows that each clause C_j contains either a positive literal x_i which has been set to *true*, or a negative literal $\neg x_i$ which has been set to *false*. Thus F is satisfied. □

In the next theorem we prove that calculating $\mathrm{I}(G)$ remains NP-hard even if G is a subcubic planar bipartite graph, which is a family of graphs not covered by Theorem 6.2.4. Moreover, it illustrates a different reduction than the one presented in the proof of Theorem 6.2.4. It is worth noting here that the gadget illustrated in Figure 6.4, which is going to be used in the proof of Theorem 6.2.7, could be used in a similar reduction as the one presented in the proof of Theorem 6.2.4 (replacing the 3-gadget) to show that calculating $\mathrm{I}(G)$ is NP-hard, when G is a subcubic bipartite graph. Nevertheless, as it will be shown in the next section, the reduction we choose to present in the proof of Theorem 6.2.7 has some direct implications on the approximability of the problem.

Theorem 6.2.7. *Let G be a graph and $k \in \mathbb{N}$. Deciding if $\mathrm{I}(G) \leq k$ is NP-complete, even if G is a subcubic planar bipartite graph.*

Proof. Since the problem is clearly in NP, we focus on proving it is also NP-hard. The reduction is from the VERTEX COVER problem, which remains NP-complete when restricted to planar cubic graphs [97]. In that problem, a planar cubic graph G and an integer $k \geq 1$ are given as an input. The question is whether there exists a vertex cover of G of order at most k. That is, whether there exists a set $VC \subseteq V(G)$ such that for every edge $uv \in E(G)$, at least one of u and v belongs to VC and $|VC| \leq k$.

Let G' be a planar cubic graph and $k \geq 1$ given as input for VERTEX COVER. Let $|E(G')| = m$. We construct a planar bipartite graph G as follows; we start with the graph G', and modify it by using multiple copies of the gadget illustrated in Figure 6.4. Note that we will be following the naming convention illustrated in Figure 6.4 whenever we talk about the vertices of our gadgets. When we say that we *attach* a copy H of the gadget to the vertices v and v' of G', we mean that we add H to G', and we identify the vertices w_1 and w_2 to the vertices v and v' respectively. Now, for every edge $vv' \in E(G')$, attach one copy H of the gadget to the vertices v and v', and then delete the edge vv' (see Figure 6.5). Clearly this construction is achieved in linear time (we have added m copies of the gadget). Note also that the resulting graph G has $\Delta(G) = 3$ and that the planarity of G' is preserved since G is constructed by essentially subdividing the edges of G' and adding a tree pending from each new vertex. Also, G is bipartite. Indeed, observe that after removing the

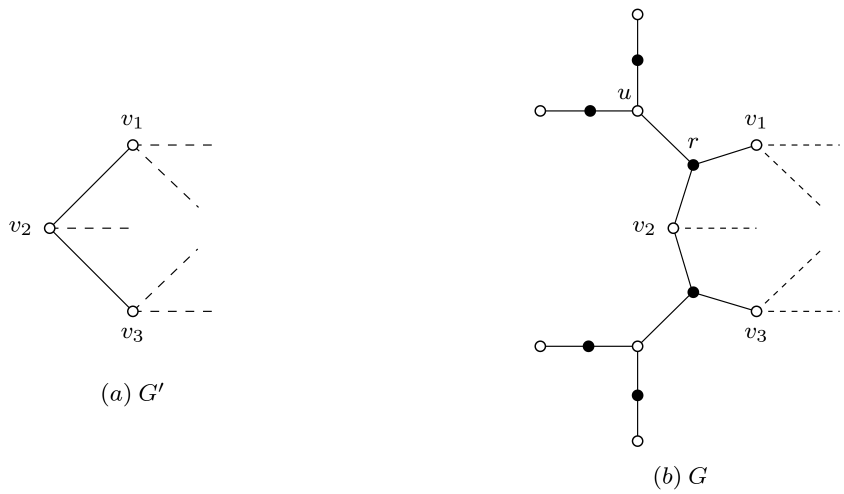


Figure 6.5 – The construction in the proof of Theorem 6.2.7. The graph G' is the initial planar cubic graph, and G is the graph built during the reduction. In G, the white and black vertices are used to denote vertices belonging to different classes of the bipartition.

edges of $E(G')$, the vertices of $V(G')$ form an independent set of G. Furthermore, the gadget is bipartite, and the vertices w_1, w_2 (that have been identified with vertices of $V(G')$) belong to the same class of the bipartition (in the gadget). Finally, for any $1 \leq i \leq m$, let H_i be the i^{th} copy of the gadget attached to vertices of G'. We will also be using the vertices r_i and u_i to denote the copies of the vertices r and u (respectively) that also belong to H_i.

We are now ready to show that the minimum vertex cover of G' has size k' if and only if $\mathrm{I}(G) = k'$. Let VC be a minimum vertex cover of G' and $|VC| = k'$. We will show that the set $S = VC$ is an irregulator of G. Let $G^* = G[V(G) \setminus S]$. Now, for any $1 \leq i \leq m$, consider the vertex r_i. Since VC is a vertex cover of G', for every edge $vv' \in E(G')$, VC contains at least one of v and v'. It follows that $d_{G^*}(r_i) \leq 2$. Note also that $N_{G^*}(r_i)$ contains the vertex $u_i \in V(H_i)$ and possibly one vertex $v \in V(G')$.

Also, since we only delete vertices in $V(H_i) \cap V(G')$, we have that $d_{G^*}(u_i) = 3 > d_{G^*}(r_i)$. In the case where $N_{G^*}(r_i)$ also contains a vertex $v \in V(G')$, the vertex v is adjacent only to vertices which do not belong to $V(G')$. Thus, $d_{G^*}(v) = d_G(v) = 3 > d_{G^*}(r_i)$. It follows that r_i has a degree that is different from that of all of its neighbours and that VC is an irregulator of G.

Now, we prove that if $\mathrm{I}(G) = k'$ then there exists a vertex cover of size at most k'. Assume that $\mathrm{I}(G) = k'$ and let S be a minimum irregulator of G. Then, S contains at least one vertex of H_i (for each $1 \leq i \leq m$). Let $X_i = V(H_i) \cap V(G')$. To construct a vertex cover VC of G' with $|VC| \leq k'$, we work as follows. For each $1 \leq i \leq m$:

1. for each vertex $v \in X_i$, if $v \in S$ then put v in VC. Then,
2. if $S \cap X_i = \emptyset$, then put any one of the two vertices of X_i in VC.

Observe now that any vertex that is added to VC during step 1. of the above procedure, also belongs to S and any vertex that is added during step 2. corresponds to at least one vertex in S. It follows that $|VC| \leq k'$. Also note that VC contains at least one vertex of X_i, for each i, and that for each $uv \in E(G')$, there exists an i such that $V(X_i) = \{u, v\}$. Thus VC is indeed a vertex cover of G'.

Therefore G' has a minimum vertex cover of size k' if and only if $\mathrm{I}(G) = k'$. To complete the proof, note that deciding if $\mathrm{I}(G) = k' < k$ for a given k, answers the question whether G' has a vertex cover of size less than k or not. ☐

6.3 (In)approximability

In the previous section we showed that computing $\mathrm{I}(G)$ is NP-hard, even for graphs G belonging to quite restricted families of graphs. So the natural question to pose next, which we investigate in this section, is whether we can approximate $\mathrm{I}(G)$.

We start with a corollary that follows from the proof of Theorem 6.2.7 and the inapproximability of VERTEX COVER in cubic graphs:

Corollary 6.3.1. *Given a graph G, it is NP-hard to approximate $\mathrm{I}(G)$ to within a ratio of $\frac{100}{99}$, even if G is bipartite and $\Delta(G) = 3$.*

Proof. The proof is the same as that of Theorem 6.2.7, where the input graph G' cubic. Therefore, the resulting graph G is a bipartite graph with $\Delta(G) = 3$ and the minimum vertex cover of G' has size k if and only if $\mathrm{I}(G) = k$. Then any a-approximation of $\mathrm{I}(G) = k$ is an a-approximation for the minimum vertex cover. Since the minimum vertex cover cannot be approximated within a factor $\frac{100}{99}$ in 3-regular graphs [54], the same holds for $\mathrm{I}(G)$. ☐

Now, we are going to show that there can be no algorithm that approximates $\mathrm{I}(G)$ to within a ratio of $\mathcal{O}(n^{1-\frac{1}{k}})$ in polynomial time, unless P=NP, even if G is a bipartite graph of order n (with no restriction on its maximum degree). One way to show that a problem can probably not be approximated within a certain ratio, is through a *gap reduction*. The goal of such a reduction is to show that it is NP-hard to differentiate between instances that have a solution of size at most α and those for which any solution has size strictly greater than β. If such is the case, then we know that we cannot approximate the optimal solution within a ratio of $\frac{\beta}{\alpha}$, as otherwise we would get that P=NP.

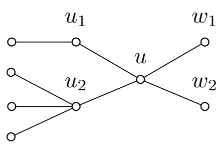
Theorem 6.3.2. *Let G be a bipartite graph of order n and $k \in \mathbb{N}$ be a constant such that $k \geq 1$. It is NP-hard to approximate $\mathrm{I}(G)$ to within $\mathcal{O}(n^{1-\frac{1}{k}})$.*

Proof. The proof is by a *gap producing reduction* from 2-BALANCED 3-SAT. Let F be a 3CNF formula with m clauses $C_1, \ldots, C_m$ and ν variables $x_1, \ldots, x_\nu$ that is given as input to the 2-BALANCED 3-SAT problem. Let $2k = k' + 1$. Based on the instance F, we are going to construct a bipartite graph $G = (V, E)$ where $|V| = \mathcal{O}(\nu^{k'+1})$ and

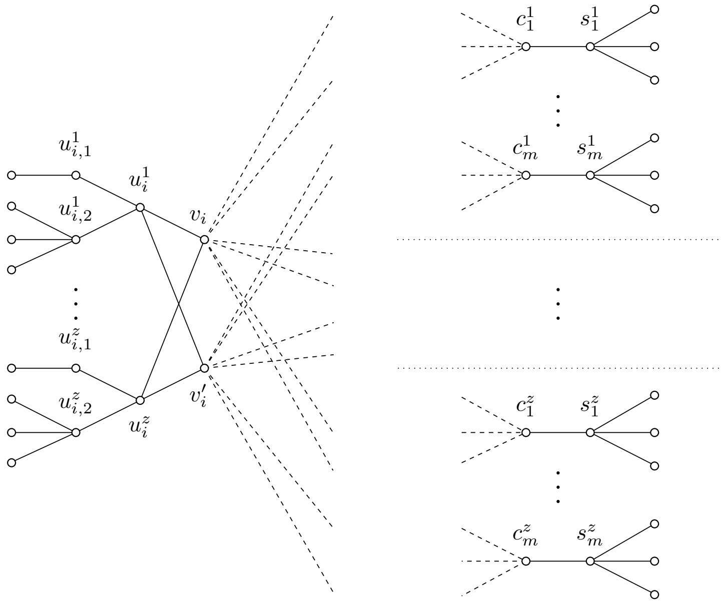
— $\mathrm{I}(G) \leq \nu$ if F is satisfiable;

— $\mathrm{I}(G) > \nu^{k'}$ otherwise.

To construct $G = (V, E)$, we start with the following graph: for each literal x_i ($\neg x_i$ resp.) in F, add a *literal vertex* v_i (v_i' resp.) in V, and for each clause C_j of F, add a *clause vertex* c_j in V. Next, for each $1 \leq j \leq m$, add the edge $v_i c_j$ ($v_i' c_j$ resp.) if the literal x_i ($\neg x_i$ resp.) appears in C_j according to F. Observe that the resulting graph is bipartite, for each clause vertex c we have $d(c) = 3$ and for each literal vertex v we have $d(v) = 2$ (since in F, each variable appears twice as a positive and twice as a negative literal). To finish the construction of G, we make use of the gadget shown in Figure 6.6 (a), as well as some copies of S_5, the star on 5 vertices. When we say that we *attach* a copy H of the gadget to the vertices v_i and v_i' (for some $1 \leq i \leq \nu$), we mean



(a) The gadget



(b) The constructed graph G for $\nu^{k'} = z$

Figure 6.6 – The construction in the proof of Theorem 6.3.2. In subfigure (b), we illustrate how each pair of literal vertices is connected to the rest of the graph. Whenever there is a superscript $1 \leq l \leq \nu^{k'}$ on a vertex, it is used to denote the l^{th} copy of that vertex. The dashed lines are used to represent the edges between the literal and the clause vertices.

that we add H to G, and we identify the vertices w_1 and w_2 to the vertices v_i and v'_i respectively. Now:

— for each $1 \leq i \leq \nu$, we attach $\nu^{k'}$ copies of the gadget to the vertices v_i and v'_i of G. For convenience, we will give unique names to the vertices corresponding to each gadget added that way. So, the vertex u^l_i (for $1 \leq l \leq \nu^{k'}$ and $1 \leq i \leq \nu$) is used to represent the vertex u of the l^{th} copy of the gadget attached to v_i and v'_i, and $u^l_{i,1}$ ($u^l_{i,2}$ resp.) is used to denote the vertex u_1 (u_2 resp.) of that same gadget. Then,

— for each $1 \leq j \leq m$, we add $\nu^{k'}$ copies of the clause vertex c_j to G, each one of these copies being adjacent to the same literal vertices as c_j. For $1 \leq l \leq \nu^{k'}$, the vertex c^l_j is the l^{th} copy of c_j. Finally,

— for each $1 \leq j \leq m$ and $1 \leq l \leq \nu^{k'}$, we add a copy of the star S_5 on five vertices to G and identify any degree-1 vertex of S_5 to c^l_j. Let s^l_j be the neighbour of c^l_j that also belongs to a copy of S_5.

Observe that the resulting graph G (illustrated in Figure 6.6 (b)) remains bipartite and that this construction is achieved in polynomial time in regards to $\nu + m$.

From the construction of G, we know that for every $1 \leq i \leq \nu$, $d(v_i) = d(v'_i) = \mathcal{O}(\nu^{k'})$. So, for sufficiently large ν, the only pairs of adjacent vertices of G that have the same degrees are either the vertices u^l_i and $u^l_{i,2}$, or the vertices c^l_j and s^l_j (for every $1 \leq i \leq \nu$, $1 \leq l \leq \nu^{k'}$ and $1 \leq j \leq m$).

First, let F be a satisfiable formula and let ϕ be a satisfying assignment of F. Also, let S be the set of literal vertices v_i (v'_i resp.) such that the corresponding literals x_i ($\neg x_i$ resp.) are assigned value *true* by ϕ. Clearly $|S| = \nu$. We will also show that S is an irregulator of G. Consider the graph $G' = G[V \setminus S]$. Now, for any $1 \leq i \leq \nu$, we have that either v_i or v'_i, say v_i, belongs to the vertices of G'. Moreover, for every $1 \leq l \leq \nu^k$, we have that $d_{G'}(u^l_i) = 3$, while $d_{G'}(u^l_{i,1}) = 2$ and $d_{G'}(u^l_{i,2}) = 4$ (since none of the neighbours of $u^l_{i,1}$ and $u^l_{i,2}$ belongs to S). Also, for every $1 \leq j \leq m$ and $1 \leq l \leq \nu^{k'}$, since ϕ is a satisfying assignment of F, $N(c^l_j)$ contains at least one vertex in S. It follows that $d_{G'}(c^l_j) = 3 < 4 = d_{G'}(s^l_j)$. Finally, since S does not contain any neighbours of v_i, we have that $d_{G'}(v_i) = d_G(v_i) = \mathcal{O}(\nu^{k'})$. It follows that S is an irregulator of G and thus that $\mathrm{I}(G) \leq \nu$.

Now let F be a non-satisfiable formula and assume that there exists an S that is an irregulator of G with $|S| \leq \nu^{k'}$. As usual, let $G' = G[V \setminus S]$. Then:

1. For every $1 \leq j \leq m$, there exists a literal vertex v such that $v \in N(c^l_j)$ for every $1 \leq l \leq \nu^{k'}$. Assume that this is not true for a specific j. Then, since $d_G(c^l_j) = d_G(s^l_j) = 4$, for every $1 \leq l \leq \nu^{k'}$, we have that S contains at least one vertex in $N[\{c^l_j, s^l_j\}]$, which does not belong to the literal vertices. That is, S contains at least one (non-literal) vertex for each one of the $\nu^{k'}$ copies of c_j. Observe also that even if this were the case, then S would also have to contain at least one more vertex to, for example, prevent $u^1_{i,2}$ and u^1_i, from having the same degree in G'. It follows that $|S| > \nu^{k'}$, which is a contradiction.
2. For every $1 \leq i \leq \nu$, S does not contain both v_i and v'_i. Assume this is not true for a specific i. Then, for every $1 \leq l \leq \nu^{k'}$, we have that $d_{G'}(u^l_i) = d_{G'}(u^l_{i,1}) = 2$, unless S also contains an additional vertex of the gadgets attached to v_i and v'_i, for each one of the $\nu^{k'}$ such gadgets. It follows that $|S| \geq \nu^{k'}$. Since we have also assumed that for a specific i, both v_i and v'_i belong to S, we have that $|S| > \nu^{k'}$, a contradiction.
3. For every $1 \leq i \leq \nu$, S contains at least one of v_i and v'_i. Assume this is not true for a specific i. Then, for every $1 \leq l \leq \nu^{k'}$, we have that $d_{G'}(u^l_i) = d_{G'}(u^l_{i,2}) = 4$, unless S also contains an additional vertex of the gadgets attached to v_i and v'_i, for each one of the $\nu^{k'}$ such gadgets. Even if this were the case, S would also have to contain at least one more vertex to, for example, prevent c^1_1 and S^1_1 from having the same degree in G'. It follows that $|S| > \nu^{k'}$, which is a contradiction.

So from items 2. and 3. above, it follows that for every $1 \leq i \leq \nu$, S contains exactly one of v_i and v'_i. Now consider the following truth assignment: we assign the value *true* to every variable x_i if the corresponding literal vertex v_i belongs to S, and value *false* to every other variable. Now, from item 1. above, it follows that each clause C_j contains either a positive literal x_i which has been set to *true*, or a negative literal $\neg x_i$ which has been set to *false*. Thus F is satisfied, which is a contradiction.

Up to this point, we have shown that there exists a graph $G = (V, E)$ with $|V(G)| = n = O(\nu^{k'+1})$ where

— $\mathrm{I}(G) \leq \nu$ if F is satisfiable;

— $\mathrm{I}(G) > \nu^{k'}$ otherwise.

Therefore, we have that there is no polynomial-time algorithm that approximates $\mathrm{I}(G)$ withing a factor of $\mathcal{O}(\nu^{k'-1})$, unless P=NP.

Now, since $n = |V(G)| = \mathcal{O}(\nu^{k'+1})$ and $2k = k' + 1$, we have $\nu^{k-1} = n^{\frac{k'-1}{k'+1}} = n^{1-\frac{2}{k'+1}} = n^{1-\frac{1}{k}}$. This ends the proof of this theorem. □

Now, we consider the case where G is a regular bipartite graph. Below we present a lower bound to the size of $\mathrm{I}(G)$. This lower bound is then used to obtain a (simple) Δ-approximation of an optimal solution.

Theorem 6.3.3. *If G is a d-regular bipartite graph $G = (L, R, E)$ of order n, we have that $\mathrm{I}(G) \geq n/2d$.*

Proof. Let S be a minimum irregulator of G and $G' = G[(L \cup R) \setminus S]$. We distinguish two cases according to if S is a subset of one of the bipartition classes L or R, or if S contains at least one vertex from each bipartition class.

Let us first deal with the first case and assume, w.l.o.g, that $S \subseteq L$. If $|S| = |L|$, then the theorem holds (since $|L| = |R|$). Therefore, we consider the case $|S| < |L|$. Observe that any vertex $v \in R$ must have $d_{G'}(v) < d$. Indeed, since $S \subseteq L$, we know that for any vertex $u \in N_{G'}(v)$, we have $d_{G'}(v) = d$ and S is an irregulator. It follows that $N(S) = R$ so $d|S| \geq n/2$ which gives us $\mathrm{I}(G) \geq n/2d$.

Now, we consider the second case (S contains at least one vertex from each bipartition class). Let $L_S = S \cap L$ and $R_S = S \cap R$. We partition L (R resp.) into three sets: $L_S = S \cap L$ ($R_S = S \cap R$ resp.), $L_{d-1} = \{u \mid u \in L \setminus S$ and $d_{G'}(u) < d\}$ ($R_{d-1} = \{u \mid u \in R \setminus S$ and $d_{G'}(u) < d\}$) and $L_d = L \setminus (L_S \cup L_{d-1})$ ($R_d = L \setminus (R_S \cup R_{d-1})$ resp.). Note that for all $u \in L_d \cup R_d$ we have $d_{G'}(u) = d$. Therefore all the vertices in L_d have exactly d neighbor in R_{d-1} (in both G and G') and all the vertices in R_d have exactly d neighbours in L_{d-1}. Furthermore, since $d_{G'}(u) < d$ for all $u \in L_{d-1} \cup R_{d-1}$, we know that each $u \in L_{d-1}$ has at least one neighbour in R_S and each $u \in R_{d-1}$ has at least one neighbour in L_S. Now we are going to find some upper bounds on the number of vertices in L_{d-1} and L_d.

Since each vertex $u \in L_{d-1}$ has at least one neighbour in R_S, we have that $|L_{d-1}| \leq d|R_S|$. Similarly we can show that $|R_{d-1}| \leq d|L_S|$.

Let E^* be the set of edges between L_d and R_{d-1}. Since, any vertex of L_d have exactly d neighbours in R_{d-1} we know that $|L_d| = |E^*|/d$. Since each vertex of R_{d-1} has at least one neighbours in L_S, it has at most $d-1$ edges in E^*. Therefore $|E^*| \leq (d-1)|R_{d-1}| \leq d(d-1)|L_S|$. This gives us that $|L_d| \leq (d-1)|L_S|$.

Now, observe that $|L| = |L_S| + |L_{d-1}| + |L_d| \leq |L_S| + d|R_S| + (d-1)|L_S| = d(|L_S| + |R_S|)$. So, since $S = L_S \cup R_S$, we have that $\mathrm{I}(G) \geq n/2d$. ☐

Now recall that in any bipartite graph G, any part of a bipartition of G into two stable sets is a vertex cover of G. Also observe that any vertex cover of a graph G is also an irregulator of G. Indeed, deleting the vertices of any vertex cover of G leaves us with an independent set, which is locally irregular. The next corollary follows from these observations and Theorem 6.3.3:

Corollary 6.3.4. *For any d-regular bipartite graph $G = (L, R, E)$, $\max\{|L|, |R|\} \leq dI(G)$ and any one of L or R is an irregulator of G.*

6.4 Parameterised complexity

So the problem of computing a minimal irregulator of a given graph G seems to be hard to solve, and even to approximate. Thus we now focus our efforts on finding parameterised algorithms that can solve it. Recall that a *fixed parameter-tractable* (FPT for short) algorithm is an algorithm with running time $f(k)n^{\mathcal{O}(1)}$, where f is a computable function and k is the considered parameter.

6.4.1 Two FPT algorithms: size of the solution/treewidth and maximum degree

First we present an FPT algorithm that calculates $\mathrm{I}(G)$ when parameterised by the size of the solution and Δ, the maximum degree of the graph. In order to prove this result, we make use of the following lemma:

Lemma 6.4.1. *Let $G = (V, E)$ be a graph such that G is not locally irregular, and S be a minimum irregulator of G. Furthermore let $G_v = (V', E')$ be the graph $G[V \setminus \{v\}]$ for a vertex $v \in S$. Then $\mathrm{I}(G_v) = \mathrm{I}(G) - 1$.*

Proof. First observe that $S' = S \setminus \{v\}$ must be an irregulator of G_v as $G_v[V' \setminus S'] = G[V \setminus S]$. It follows that $\mathrm{I}(G_v) \leq \mathrm{I}(G) - 1$. Assume that $\mathrm{I}(G_v) < \mathrm{I}(G) - 1$. Then there exists an S'' such that $|S''| < \mathrm{I}(G) - 1$ and S'' is an irregulator of G_v. Since $G_v[V' \setminus S''] = G[V \setminus (S'' \cup \{v\})]$, we have that $S'' \cup \{v\}$ is an irregulator of G and $|S'' \cup \{v\}| = |S''| + 1 < \mathrm{I}(G)$. This is a contradiction. □

We are now ready to present the proof of the following theorem:

Theorem 6.4.2. *For a given graph $G = (V, E)$ with $|V| = n$ and maximum degree Δ, and for $k \in \mathbb{N}$, there exists an algorithm that decides if $\mathrm{I}(G) \leq k$ in time $(2\Delta)^k n^{\mathcal{O}(1)}$.*

Proof. In order to decide if $\mathrm{I}(G) \leq k$ we are going to use a recursive algorithm. The algorithm takes (G, k) as input, where $G = (V, E)$ is a graph and $k \geq 0$ is an integer. The basic idea of this algorithm is to take advantage of Observation 6.1.5. We present the exact procedure in Algorithm IsIrregular.

Input: A graph $G = (V, E)$ and an integer $k \geq 0$.
Output: Is $\mathrm{I}(G) \leq k$ or not?
1: **if** G is irregular **then**
2:     **return** yes
3: **else if** $k = 0$ **then**
4:     **return** no
5: **else** ▷ $k > 0$ and G is not irregular
6:     $ans \leftarrow no$
7:     find an edge $vu \in E$ such that $d_G(v) = d_G(u)$
8:     **for all** $w \in N_G[\{u, v\}]$ **do**
9:         set $G_w = G[V \setminus \{w\}]$
10:         **if** IsIrregular($G_w, k - 1$) returns yes **then**
11:             $ans \leftarrow yes$
12:     **return** ans

Algorithm 6.1 – [IsIrregular(G, k) decision function]

Now, let us argue about the correctness and the efficiency of this algorithm. We claim that for any graph $G = (V, E)$ and any integer $k \geq 0$, Algorithm 6.1 returns yes if $\mathrm{I}(G) \leq k$ and no otherwise. Furthermore, the number of steps that the algorithm requires is $f(k, n) = (2\Delta)^k n^{\mathcal{O}(1)}$, where $n = |V|$. We prove this by induction on k.

Let us first deal with the base of the induction ($k = 0$). Here, we only need to check if G is locally irregular. Algorithm 6.1 does this in line 1 and returns yes if it is (line 2) and no otherwise (line 4). Furthermore, we can check if G is locally irregular in polynomial time. So, the claim is true for the base case.

The induction hypothesis is for ($k = k_0 \geq 0$). We assume that we have a $k_0 \geq 0$ such that Algorithm 6.1 can decide if any graph G with n vertices and maximum degree Δ has $\mathrm{I}(G) \leq k_0$ in $f(k_0, n) = (k_0 + 1)(2\Delta)^{k_0} n^{\mathcal{O}(1)}$ steps.

Let us now show the induction step ($k = k_0 + 1$). Let $G = (V, E)$ be a graph. If G is locally irregular then $\mathrm{I}(G) = 0$ and Algorithm 6.1 answers correctly (in line 2). Assume that G is not locally irregular; then there exists an edge $vu \in E$ such that $d_G(v) = d_G(v)$. Now, let S be a minimum irregulator of G. It follows from Observation 6.1.5 that S must include at least one vertex $w \in N_G[\{v, u\}]$. Since Algorithm 6.1 considers all the vertices in $N_G[\{v, u\}]$, at some point it also considers the vertex $w \in S \cap N_G[\{v, u\}]$. Now, observe that for any $x \in S$, the set $S_x = S \setminus \{x\}$ is a minimum irregulator of G_x, where $G_x = G[V \setminus \{x\}]$. Furthermore, by Lemma 6.4.1, we have $\mathrm{I}(G_x) \leq k - 1 = k_0$ if and only if $\mathrm{I}(G) \leq k$. By the induction hypothesis, we know that the algorithm answers correctly for all the instances (G_x, k_0). Thus, if $\mathrm{I}(G) \leq k = k_0 + 1$, there must exist one instance (G_w, k_0), where $w \in S \cap N_G[\{v, u\}]$, for which Algorithm 6.1 returns yes. Therefore the algorithm answers for $(G, k_0 + 1)$ correctly. Finally, this process requires $n^{\mathcal{O}(1)}$ steps in order to check if the graph is locally irregular and $2\Delta f(k-1, n-1)$ steps (by induction hypothesis) in order to check if for any graph G_x we have $\mathrm{I}(G_x) \leq k - 1 = k_0$ (where $x \in N[\{u, v\}]$). So, the algorithm decides in $n^{\mathcal{O}(1)} + 2\Delta f(k - 1, n - 1) \leq n^{\mathcal{O}(1)} + 2\Delta k(2\Delta)^{k-1}(n-1)^{\mathcal{O}(1)} \leq n^{\mathcal{O}(1)} + k(2\Delta)^k n^{\mathcal{O}(1)} \leq (k+1)(2\Delta)^k n^{\mathcal{O}(1)}$ steps. The result follows from the fact that $k \leq n - 1$. □

We now turn our attention towards graphs of bounded treewidth. In particular, we provide an FPT algorithm that finds a minimum irregulator of G when parameterised by the treewidth of the input graph and by Δ. Recall the basic notions and notations explained in Section 2.2.2.

Theorem 6.4.3. *For a given graph $G = (V, E)$ and a nice tree-decomposition of G, there exists an algorithm that returns $\mathrm{I}(G)$ in time $\Delta^{4tw} n^{\mathcal{O}(1)}$, where tw is the treewidth of the given decomposition and Δ is the maximum degree of G.*

Proof. We are going to perform dynamic programming on the nodes of the given nice tree-decomposition. The idea behind our algorithm is that for each node t we store all the sets $S \subseteq V_t$ such that S is an irregulator of $V_t \setminus X_t$ in G. We will also store the necessary "conditions" (explained more in what follows) such that if there exists a set S', where $S' \setminus S \subseteq V \setminus V_t$, that meets these conditions, then S' is an irregulator of V_t in G. Observe that if we manage to do such a thing for every node of the tree-decomposition, then we can find $\mathrm{I}(G)$. To do so, it suffices to check the size of all the irregulators we stored for the root r of the tree-decomposition, which also meet the conditions we have set. In that way, we can find a set S that is an irregulator of $V_r \setminus X_r$ in G, satisfies our conditions and is of minimum order. Since $V_r = V$ and $X_r = \emptyset$, this set S is a minimum irregulator of G and $\mathrm{I}(G) = |S|$.

Let us now present the actual information we are keeping for each node. Assume that t is a node of the tree-decomposition and $S \subseteq V_t$ is an irregulator of $V_t \setminus X_t$ in G, *i.e.*, S is an irregulator of $V_t \setminus X_t$ in G. For this S we want to remember which vertices of X_t belong to S as well as the degrees of the vertices $v \in X_t \setminus S$ in $G[V_t \setminus X_t]$. This can be done by keeping a table D of size $tw + 1$ where, if $v \in X_t \setminus S$, then we set $D(v) = d_{G[V_t \setminus X_t]}(v)$ and if $v \in X_t \cap S$, then we set $D(v) = \emptyset$ (slightly abusing the notation, by $D(v)$ we mean the position in the table D that corresponds to the vertex v). Like we have already said, we are going to keep some additional information about the conditions that could allow these sets to be extended to irregulators of V_t in G if we add vertices of $V \setminus V_t$. For that reason, we are also going to keep a table with the "*target degree*" of each vertex; in this table we assign to each vertex $v \in X_t \setminus S$ a degree d_v such that, if there exists S' where $S' \setminus S \subseteq V \setminus V_t$ and for all $v \in X_t \setminus S$ we have $d_{G[V \setminus S']}(v) = d_v$, then S is an irregulator of V_t in G. This can be done by keeping a table T of size $tw + 1$ where for each $v \in X_t \setminus S$ we set $T(v) = i$, where i is the target degree, and for each $v \in X_t \cap S$ we set $T(v) = \emptyset$. Such tables T will be called *valid* for S in X_t. Finally, we are going to keep the set $X = S \cap X_t$ and the value $min = |S|$. Note that the set X does not give us any extra information, but we keep it as it will be useful to refer to it directly.

To sum up, for each node t of the tree-decomposition of G, we keep a set of quadruples (X, D, T, min), each quadruple corresponding to a valid combination of a set S that is an irregulator of $V_t \setminus X_t$ in G and the target degrees for the vertices of $X_t \setminus S$. Here it is important to say that when treating the node X_t, for every two quadruples (X_1, D_1, T_1, min_1) and (X_2, D_2, T_2, min_2) such that for all $v \in X_t$ we have that $D_1(v) = D_2(v)$ and $T_1(v) = T_2(v)$ (this indicates that $X_1 = X_2$ as well), then we are only going to keep the quadruple with the minimum value between min_1 and min_2 as we will prove that this is enough in order to find $\mathrm{I}(G)$.

Claim 6.4.4. *Assume that for a node t, we have two sets S_1 and S_2 that are both irregulators of $V_t \setminus X_t$ in G, and that T is a target table that is common to both of them. Furthermore, assume that $(X_1, D_1, T, |S_1|)$ and $(X_2, D_2, T, |S_2|)$ are the quadruples we have to store for S_1 and S_2 respectively (both respecting T), with $D_1(v) = D_2(v)$ for every $v \in X_t$. Then for any set $S \subseteq V \setminus V_t$ such that $d_{G[V \setminus (S_1 \cup S)]}(v) = T(v)$ for all $v \in X_t$, we also have that $d_{G[V \setminus (S_2 \cup S)]}(v) = T(v)$ for all $v \in X_t$.*

Proof of the claim. Assume that we have such an S for S_1, let v be a vertex in X_t and $H = G[v \cup ((V \setminus V_t) \setminus S)]$ (observe that H does not depend on S_1 or S_2). Since $d_{G[V \setminus (S_1 \cup S)]}(v) = T(v)$, we know that in the graph H, v has exactly $T(v) - D_1(v)$ neighbours (as $D_1(v) = d_{G[V_t \setminus S_1)]}(v)$). Now, since $D_1(v) = D_2(v) = d_{G[V_t \setminus S_2]}(v)$ we have that $d_{G[V \setminus S_2 \cup S]}(v) = T(v)$. Therefore, the claim holds. ◇

Simply put, Claim 6.4.4 states that for any two quadruples $Q_1 = (X, D, T, min_1)$ and $Q_2 = (X, D, T, min_2)$, any extension S of S_1 is also an extension of S_2 (where S_1 and S_2 are the two sets that correspond to Q_1 and Q_2 respectively). Therefore, in order to find the minimum solution, it is sufficient to keep the quadruple that has the minimum value between min_1 and min_2.

Now we are going to explain how we create all the quadruples (X, D, T, min) for each type of node in the tree-decomposition.

First, let t be a leaf node. Observe that in this case, $X_t = V_t = \emptyset$. Therefore, we have only one quadruple (X, D, T, min), where the size of both D and T is zero (so we do not need to keep any information in them), $S = \emptyset$ and $min = |S| = 0$.

Let t be an introduce node; assume that we have all the quadruples (X, D, T, min) for its child t' and let v be the introduced vertex. By construction, we know that v is introduced in X_t and thus it has no neighbours in $V_t \setminus X_t$. It follows that if $S \subseteq G_{t'}$ is an irregulator for $G_{t'} \setminus X_{t'}$, then both S and $S \cup \{v\}$ are irregulators for $V_t \setminus X_t$ in G. Furthermore, there is no set $S \subseteq V_t \setminus \{v\}$ that is an irregulator of $V_t \setminus X_t$ and is not an irregulator of $G_{t'} \setminus X_{t'}$. So, we only need to consider two cases for the quadruples we have to store for c; if v belongs to the under-construction irregulator of $V_t \setminus X_t$ in G or not.

Case 1. (v is in the irregulator): Observe that for any S that is an irregulator of $G_{t'} \setminus X_{t'}$ in G, which is stored in the quadruples of $X_{t'}$, for every $u \in X_{t'} \setminus S$, we have that $d_{G[G_{t'} \setminus S]}(u) =$

$d_{G[V_t \setminus (S \cup \{v\})]}(u)$. Moreover, for any target table T which is valid for S in t', the target table T' is valid for $S \cup \{v\}$ in t, where T' is almost the same as T, the only difference being that T' also contains the information about v, *i.e*, $T'(v) = \emptyset$. So, for each quadruple (X, D, T, min) in t', we need to create one quadruple $(X \cup \{v\}, D', T', min + 1)$ for t, where D' is almost the same as D, except that it also contains the information about v, *i.e.*, $D'(v) = \emptyset$.

Case 2. (v is not in the irregulator): Let $q = (X, D, T, min)$ be a stored quadruple of t' and S be the corresponding irregulator of $G_{t'} \setminus X_{t'}$ in G. We first explain how to construct D' of t, based on q. Observe that the only change between $G[G_{t'} \setminus S]$ and $G[V_t \setminus S]$ is that in the latter there exist some new edges from v to some of the vertices of $X_{t'}$. Therefore, for each vertex $u \in X_{t'} \setminus X$ we set $D'(u) = D(u) + 1$ if $u \in N[v]$ and $D'(u) = D(u)$ otherwise. Finally, for the introduced vertex v, we set $D'(v) = |N(v) \cap (X_{t'} \setminus X)|$. We now treat the target degrees for t. Observe that the target degrees for each vertex in $X_t \setminus \{v\}$ are the same as in T, since v only has edges incident to vertices in X_t. Now, we only need to decide which are the valid targets for v. Since $d_{G[V_t \setminus S]}(v) = D'(v)$, we know that for every target τ, we have that $D'(v) \leq \tau \leq \Delta$. Furthermore, we cannot have the target degrees of v be the same as the targets of one of its neighbours in $X_{t'}$ (these values are stored in T), as, otherwise, any valid target table T' of t would lead to adjacent vertices in X_t having the same degree. Let $\{\tau_1, \ldots, \tau_k\} \subset \{D(v), \ldots, \Delta\}$ be an enumeration of all the valid targets for v (i.e. $\tau_i \neq T(u)$ for all $u \in N[v] \cap X_{t'} \setminus X$). Then, for each quadruple (X, D, T, min) in t', and for each $i = 1, \ldots, k$, we need to create the quadruple (X, D', T_i, min), such that $T_i(u) = T(u)$ for all $u \in X_{t'}$ and $T_i(v) = \tau_i$. In total, we have $k \leq \Delta$ such quadruples.

Let t be a join node with t' and t'' as its two children in the tree-decomposition. Here, it is important to mention that $X_{t'} = X_{t''}$ and $(V_{t'} \setminus X_{t'}) \cap (V_{t''} \setminus X_{t''}) = \emptyset$. Assume that there exists an irregulator S of $V_t \setminus X_t$ in G, a valid target table T of S, and let (X, D, T, min) be the quadruple we need to store in t for this pair (S, T). Observe that this pair (S, T) is valid for both t' and t'', so we must already have stored at least one quadruple in each node. Let $X \subseteq X_t$ and a let T be a target table such that (X, D_1, T, min_1) and (X, D_2, T, min_2) are stored for t' and t'' respectively. We create the quadruple (X, D, T, min) for t by setting $D(u) = D_1(u) + D_2(u) - d_{G[X_t \setminus X]}(u)$ for all $u \in X_t \setminus X$, $D(u) = \emptyset$ for all $u \in X$ and $min = min_1 + min_2 - |X|$. Observe that these are the correct values for the $D(u)$ and min, as otherwise we would count $d_{G[X_t \setminus X]}(u)$ and $|X|$ twice. Finally, we need to note that we do not store any quadruple (X, D, T, min) we create for the join node such that $D(u) > T(u)$ for a vertex $u \in X_t \setminus X$. This is because for such quadruples, the degree of vertex u will never be equal to any of the target degrees we have set, as it can only increase when we consider any of the ancestor (*i.e.* parent, grantparent etc.) nodes of t.

Finally, let t be a forget node, t' be its child and v be the forgotten vertex. Assume that we have to store in t a quadruple (X, D, T, min). Then, since $X = X_t \cap S$ for an irregulator S of X_t in G, we know that in t' we must already have stored a quadruple (X', D', T', min') such that $X' = S \cap X_{t'}$, $D'(u) = D(u)$ for all $u \in X_{t'}$, $T'(u) = T(u)$ for all $u \in X_{t'}$ and $min' = min$. Therefore, starting from the stored quadruples in t', we can create all the quadruples of t. For each quadruple (X', D', T', min') in t', we create at most one quadruple (X, D, T, min) for t by considering two cases; the forgotten vertex v belongs to X' or not.

Case 1. *(v belongs to X')*: then the quadruple (X, D, T, min) is almost the same as (X', D', T', min'), with the following differences: $X = X' \setminus \{v\}$, $min = min'$, $D(u) = D'(u)$ and $T(u) = T'(u)$ for all $u \in X_t$ and the tables D and T do not include any information for v as this vertex does not belong to X_t anymore.

Case 2. *(v does not belong to X')*: we first check if $D'(v) = T'(v)$ or not. This is important because the degree of v will never again be considered by our algorithm, and thus its degree will remain unchanged. So, if $D'(v) = T'(v)$, we create the quadruple (X, D, T, min) where $X = X'$, $min = min'$, $D(u) = D'(u)$ and $T(u) = T'(u)$ for all $u \in X_t$ and the tables D and T do not include any information for v.

For the running time, observe that the number of nodes of a nice tree-decomposition is $\mathcal{O}(tw \cdot n)$ and all the other calculations are polynomial in $n+m$. Thus we only need to count the different quadruples in each node. Now, for each vertex v, we either include it in X or we have $\Delta+1$ options for the value $D(u)$ and $\Delta+1-i$ for the value $T(u)$ if $D(u) = i$. Also, for sufficiently large Δ, we have that $1+\sum_{i=0}^{\Delta}(\Delta+1-i) < \Delta^2$. Furthermore, the set X and the value min do not increase the number of quadruples because $X = \{u \mid D(u) = \emptyset\}$ and from all quadruples (X, D_1, T_1, min_1), (X, D_2, T_2, min_2) such that $D_1(u) = D_2(u)$ and $T_1(u) = T_2(u)$ for all $u \in X_t$, we only keep one of them (by Claim 6.4.4).

In total, there are Δ^{2tw} different quadruples in each node, and, taking in account the combinations needing to be checked for the join nodes, the algorithm decides in $\Delta^{4tw}n^{\mathcal{O}(1)}$ time. □

It is worth noting that the algorithms of Theorems 6.4.2 and 6.4.3 can be used in order to also return a minimum irregulator of G. Moreover, for a given graph G, the problem of calculating $\mathrm{I}(G)$ is in XP when parameterised only by the size of the solution or by the treewidth of G (without considering $\Delta(G)$ as part of the parameter in either case).

6.4.2 W-hardness

Observe that both of the algorithms presented above have to consider Δ as part of the parameter if they are to be considered as FPT. The natural question to ask at this point is whether we can have an FPT algorithm parameterised only by the size of the solution, or by the treewidth of the input graph. In this section, we give a strong indication towards a negative answer for both cases, proving that, in some sense, the algorithms provided in Section 6.4.1 are optimal. To achieve that, we also make use of what is known as a *linear fpt-reduction*, a type of polynomial reduction such that the size of the parameter of the new problem is linear in regards to the size of the parameter of the original problem. Observe that if we have a linear fpt-reduction from a problem Q with parameter k to a problem Q' with parameter k' and the assumption that Q cannot be solved in time $f(k)n_1^{o(k)}$ (where n_1 is the size of the input of Q), then we can conclude that there is no $f(k')n_2^{o(k')}$-time algorithm for Q (where n_2 is the size of the input of Q).

Theorem 6.4.5. *Let G be a graph and $k \in \mathbb{N}$. Deciding if $\mathrm{I}(G) \leq k$ is $\mathrm{W}[2]$-hard, when parameterised by k.*

Proof. The reduction is from the DOMINATING SET problem, which was shown to be $\mathrm{W}[2]$-complete when parameterised by the size of the solution (*e.g.* in [57]). In that problem, a graph $H = (V, E)$ and an integer k are given as input. The question asked is whether there exists a set $D \subseteq V$ of order at most k (called a *dominating set of H*) such that $V = N[D]$.

Let $H = (V, E)$ be a graph and $k \in \mathbb{N}$. We construct a graph $G = (V', E')$ such that H has a dominating set of order at most k if and only if G has an irregulator of order at most k. We begin by setting $V = \{v_1, \dots, v_n\}$. The graph G is built starting from a copy of the graph H. To avoid any confusion in what is to follow, we will always use H to denote the original graph, and $G|_H = G[\{v'_1, \dots, v'_n\}]$ to denote the copy of H that lies inside G (where the indices of the v'_is are the same as the indices of the corresponding v_is). Then, for each $1 \leq i \leq n$, we attach the necessary number of pending vertices (meaning vertices of degree 1) to the vertex v'_i, so that the degree of v'_i becomes equal to $i \cdot n$. Finally, for each v'_i, let u'_i be one of its newly attached pending vertices, and attach the necessary number of new pending vertices to u'_i, so that its degree becomes equal to that of v'_i. The resulting graph is G. To be clear, for every vertex v of G, we either have that $v = v'_i$ or $v = u'_i$, or that v is a vertex pending from v'_i or u'_i (for some $1 \leq i \leq n$). Note also that for each $1 \leq i \leq n$, we have that $d_G(v'_i) = d_G(u'_i) = i \cdot n$.

Now let D be a dominating set of H, with $|D| = m \leq k$, and let D' be the subset of V' that corresponds to the vertices of D. That is, $D' = \{v'_i \in V' : v_i \in D\}$. We claim that the graph $G' = G[V' \setminus D']$ is locally irregular. Indeed, for every $1 \leq i \leq n$, let $\alpha(i)$ be the number of neighbours of v_i that belong to D. Observe that since D is a dominating set of H, we have that $1 \leq \alpha(i) \leq n-1$. Now, for every vertex v'_i in V', we have that either $v'_i \in D'$, in which case v'_i does not belong to G', or $d_{G'}(v'_i) = d_G(v'_i) - \alpha(i) < d_{G'}(u'_i)$. Moreover, for every $1 \leq i < j \leq n$, if $v'_i, v'_j \notin D'$, we have that $d_G(v'_j) - d_G(v'_i) \geq n$, and thus $d_{G'}(v'_j) - d_{G'}(v'_i) = d_G(v'_j) - \alpha(j) - d_G(v'_i) + \alpha(i) \geq n + \alpha(i) - \alpha(j) \geq 2$. Finally, every pending vertex l of G' is attached to either u'_i or v'_i, which have degree (in G') strictly larger than 1. It follows that D' is an irregulator of G with $|D'| = m \leq k$, and thus $\mathrm{I}(G) \leq k$.

For the other direction, assume that $\mathrm{I}(G) \leq k$ and let S be an irregulator of G, with $|S| = k$, and $G' = G[V' \setminus S]$. For each $1 \leq i \leq n$, let $S_i = N[v'_i] \cup N(u'_i)$. We claim that for every i, we have $S \cap S_i \neq \emptyset$. Assume that this is not true, *i.e.*, that there exists an i_0 such that $S_{i_0} \cap S = \emptyset$. Then, by deleting the vertices of S from G, the degrees of v'_{i_0} and u'_{i_0} remain unchanged. Formally, we have that $d_{G'}(v'_{i_0}) = d_G(v'_{i_0}) = d_G(u'_{i_0}) = d_{G'}(u'_{i_0})$. This is a contradiction since S is an irregulator of G. Now, we consider the set S', defined as follows:

— Start with $S' = S$.
— For each i, while there exists a vertex $v \in S_i \cap S'$ such that $d_G(v) = 1$ or $v = u'_i$, remove v from S' and add v'_i to S'.

Clearly, we have that S' only contains vertices from $V(G|_H)$ and that $|S'| \leq |S| = k$. Also, from the construction of S', for every i, we have that $S_i \cap S' \neq \emptyset$. It follows that for every vertex v'_i, we either have $v'_i \in S'$ or there exists a vertex $v \in N(v'_i) \cap V(G|_H)$ such that $v \in S'$. Going back to H, let $D = \{v_i : v'_i \in S'\}$. It is clear that D is a dominating set of H of order at most k. This finishes our reduction.

Finally, note that throughout the above described reduction, the value of the parameter of the two problems is the same (in both of them, the parameter has value k). Moreover, the construction of the graph G is achieved in polynomial time in regards to n. These observations conclude our proof. □

Theorem 6.4.6. *Let G be a graph with treewidth tw, and $k \in \mathbb{N}$. Deciding if $\mathrm{I}(G) = k$ is $W[1]$-hard when parameterised by tw.*

Proof. We will present a reduction from the LIST COLOURING problem: the input consists of a graph $H = (V, E)$ and a list function $L : V \to \mathcal{P}(\{1, \dots, k\})$ that specifies the available colours for each vertex $u \in V$. The goal is to find a proper colouring $c : V \to \{1, \dots, k\}$ such that $c(u) \in L(u)$ for all $u \in V$. When such a colouring exists, we say that (H, L) is a *yes-instance* of LIST COLOURING. This problem is known to be $W[1]$-hard when parameterised by the treewidth of H [61].

Now, starting from an instance (H, L) of LIST COLOURING, we construct a graph $G = (V', E')$ (see Figure 6.7 (a)) such that:

— $|V'| = \mathcal{O}(|V|^6)$,
— $tw(G) = tw(H)$ and
— $\mathrm{I}(G) = nk$ if and only if (H, L) is a yes-instance of LIST COLOURING.

Before we start with the construction of G, let us present the following claim.

Claim 6.4.7. *Let (H, L) be an instance of* LIST COLOURING *where $H = (V, E)$ and there exists a vertex $u \in V$ such that $|L(u)| > d(u)$. Then the instance $(H[V \setminus \{u\}], L')$, where $L'(v) = L(v)$ for all $v \in V \setminus \{u\}$, is a yes-instance of* LIST COLOURING *if and only if (H, L) is a yes-instance of* LIST COLOURING.

Proof of the claim. Indeed, observe that for any vertex $u \in V$, by any proper colouring c of H, $c(u)$ only has to avoid $d(u)$ colours. Since $|L(u)| > d(u)$, we will always have a spare colour to use on u that belongs to $L(u)$. $\diamond$

From the previous claim, we can assume that in our instance, for all $u \in V$, we have $|L(u)| \leq d(u)$. Furthermore, we can deduce that $k \leq n(n-1)$ as the degree of any vertex is at most $n - 1$. Finally, let us denote by $\overline{L}(u)$ the set $\{0, 1, \ldots, k\} \setminus L(u)$. It is important to note here that for every $u \in V$, the list $L(u)$ contains at least one element belonging in $\{1, \ldots, k\}$. It follows that $\overline{L}(u)$ also contains at least one element, the colour 0. To sum up, we have that $1 \leq |\overline{L}(u)| \leq k$.

Now, we present the three gadgets we are going to use in the construction of G. First, we have the "forbidden colour gadget" H_i, which is a star with i leaves (see Figure 6.7 (c)). When we say that we attach a copy of H_i on a vertex v of a graph G, we mean that we add H_i to G and we identify the vertices v and w_2 (where here and in what follows, we are using the naming illustrated in Figure 6.7 when talking about the vertices w_1, w_2, w_3, v_1 and v_2). The second is be the "degree gadget", which is presented in Figure 6.7 (b). Finally, we have the "horn gadget", which is a path on three vertices (see Figure 6.7 (d)). We define the operation of attaching these two gadgets on a vertex v of a graph G similarly to how we defined this operation for the forbidden colour gadget (each time using the appropriate w_1 or w_3, according to if it is a degree or a horn gadget respectively).

In order to construct G, we start from a copy of H. Let us use $G|_H$ to denote the copy of H that lies inside of G and, for each vertex $u \in V$, let u' be its copy in V'. We will call the set of these vertices U. That is, $U = \{v \in V(G|_H)\}$. Then, we are going to attach several copies of each gadget to u', for each vertex $u' \in U$. We start by attaching k copies of the degree gadget to each vertex $u' \in U$. Then, for each $u \in V$ and each $i \in \overline{L}(u)$, we attach one copy of the forbidden colour gadget H_{2n^3-i} to the vertex u'. Finally, for each $u' \in U$, we attach to u' as many copies of the horn gadget as are needed, in order to have $d_G(u') = 2n^3$.

Before we continue, observe that, for sufficiently large n, we have attached more than n^3 (but still a polynomial number) horn gadgets to each vertex of U. Indeed, before attaching the horn gadgets, each vertex $u' \in U$ has $d_G(u) \leq n - 1$ neighbours in U, k neighbours from the degree gadgets and at most $k < n^2$ neighbours from the forbidden colour gadgets (recall that $|\overline{L}(u)| \leq k$). We now show that $|V'| = \mathcal{O}(n^6)$. For that purpose, let us calculate the number of vertices in all the gadgets attached to a single vertex $u' \in U$. First, we have $5k < 5n^2$ vertices in the degree gadgets. Then, we have less than $4n^3$ vertices in the horn gadgets (as we have less that $2n^3$ such gadgets). Finally, we have at most $k < n^2$ forbidden colour gadgets, each one of which contains at most $2n^3$ vertices. So, for each vertex $u' \in U$, we have at most $2n^5 + 4n^3 + 5n^2$ vertices in the gadgets attached to u'. Therefore, we have $|V'| = \mathcal{O}(n^6)$.

Before we prove that $\mathrm{I}(G) \leq nk$ if and only if (H, L) is a yes-instance of LIST COLOURING, we need to argue about two things. First, about the treewidth of the graph G and second about

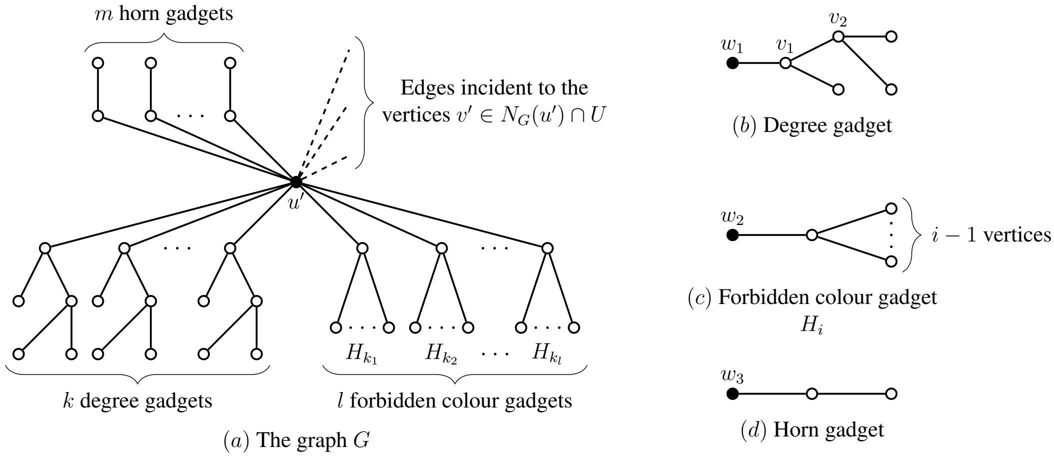


Figure 6.7 – In (a) we illustrate the construction of G, as it is described in the proof of Theorem 6.4.6. The black vertex represents every vertex that belongs to U. For the specific vertex u' shown in the figure, we have that $\overline{L}(u) = \{c_1, \ldots, c_l\}$ and $k_i = n^3 - c_i$ for all $i = 1, \ldots, l$. We also have that $m = 2n^3 - d_G(u) - k - l$.

the minimum value of $\mathrm{I}(G)$. Since our construction only attaches trees to each vertex of $G|_H$ (and recall that a tree has a treewidth of 1 by definition), we know that $tw(G) = tw(G|_H) = tw(H)$. As for $\mathrm{I}(G)$, we will show that it has to be at least equal to nk. For that purpose we have the following two claims.

Claim 6.4.8. *Let S be an irregulator of G and $S \cap U \neq \emptyset$. Then $|S| > n^3$.*

Proof of the claim. Let $u' \in S \cap U$. By construction, G contains more than n^3 horn gadgets that are attached to u'. Therefore, by deleting u', we create more than n^3 copies of the graph P_2, each one of which forces us to include at least one of its vertices in S. Hence, $|S| > n^3$. $\diamond$

Claim 6.4.9. *Let S be an irregulator of G and $S \cap U = \emptyset$. Then $|S| \geq nk$. In particular, S includes at least one vertex from each copy of the degree gadget used in the construction of G.*

Proof of the claim. Let D be a copy of the degree gadget, attached to some vertex $u' \in U$. Observe that we have $d_G(v_1) = d_G(v_2)$. It follows by Observation 6.1.5 that S contains at least one vertex v in $N[\{v_1, v_2\}]$, and since $u' \notin S$, this v is a vertex other than w_1. The result follows from the fact that the same arguments hold for any degree gadget attached to any vertex of U (recall that $|U| = n$ and we have attached k copies of the degree gadget to each one of the vertices of U). Hence, $|S| \geq nk$. $\diamond$

By the previous two claims, we conclude that $\mathrm{I}(G) \geq nk$. We are ready to show that if (H, L) is a yes-instance of LIST COLOURING, then there exists a set $S \subseteq V'$ such that S is an irregulator of G and $|S| = nk$. Let c be a proper colouring of H such that $c(u) \in L(u)$ for all $u \in V$. We construct an irregulator of G as follows. For each $u \in V$, we partition (arbitrarily) the k degree gadgets attached to the vertex u' to $c(u)$ "good" and $(k - c(u))$ "bad" degree gadgets. For each good degree gadget, we add the copy of the vertex v_1 of that gadget to S and for each bad degree gadget we add the copy of the vertex v_2 of that gadget to S. This process creates a set S of size nk, as it includes k distinct vertices for each vertex $u' \in U$.

Now we need to show that S is an irregulator of G. Let $G' = G[V' \setminus S]$; observe that each vertex $u' \in U$ has degree $d_{G'}(u') = 2n^3 - c(u)$. Therefore, u' does not have the same degree as any of its neighbours that do not belong in U. Indeed, for every $v \in N_{G'}(u') \setminus U$, we have that $d_{G'}(v) \in \{1, 2\}$ (if v belongs to a bad degree or a horn gadget) or $d_{G'}(v) \in \{2n^3 - i : i \in \overline{L}(u)\}$ (if v belongs to a forbidden colour gadget). Furthermore, since c is a proper colouring of H, for all $uv \in E$, we have that $c(u) \neq c(v)$. This gives us that for any edge $u'v' \in E'$ with $u', v' \in U$, we have that $d_{G'}(u') = 2n^3 - c(u) \neq 2n^3 - c(v) = d_{G'}(v')$.

So, we know that for every vertex $u' \in U$, there is no vertex $w \in N_{G'}(u')$ such that $d_{G'}(u') = d_{G'}(w)$. It remains to show that, in G', there exist no two vertices belonging to the same gadget which have the same degree. First of all, we have that S does not contain any vertex from any of the horn and forbidden colour gadgets, nor from U. Thus any adjacent vertices belonging to these gadgets have different degrees. Last, it remains to check the vertices of the degree gadgets. Observe that for any copy of the degree gadget, S contains either v_1 or v_2. In both cases, after the deletion of the vertices of S, any adjacent vertices belonging to any degree gadget have different degrees. Therefore, S is an irregulator of G of order nk and since $\mathrm{I}(G) \geq nk$ we have that $\mathrm{I}(G) = nk$.

Now, for the opposite direction, assume that there exists a set $S \subseteq V'$ such that S is a minimum irregulator of G and $|S| = nk$. Let $G' = (V'', E'')$ be the graph $G[V' \setminus S]$. It follows from Claims 6.4.8 and 6.4.9, that $S \cap U = \emptyset$ and that S contains exactly one vertex from each copy of the degree gadget in G and no other vertices. Consider now the colouring c of H defined as $c(u) = 2n^3 - d_{G'}(u')$. We show that c is a proper colouring for H and that $c(u) \in L(u)$. First, we have that c is a proper colouring of H. Indeed, for any edge $uv \in E$, there exists an edge $u'v' \in E''$ (since $S \cap U = \emptyset$). Since G' is locally irregular we have that $d_{G'}(u') \neq d_{G'}(v')$, and thus $c(u) \neq c(v)$. It remains to show that $c(u) \in L(u)$ for all $u \in V$. First observe that, during the construction of G, we attached exactly k degree gadgets to each $u' \in U$. It follows that $d_{G'}(u') = 2n^3 - j$ and $c(u) = j$ for a $j \in \{0, 1, \dots, k\}$. It is sufficient to show that $j \notin \overline{L}(u)$. Since S contains only vertices from the copies of the degree gadgets, we have that each $u' \in U$ has exactly one neighbour of degree $2n^3 - i$ for each $i \in \overline{L}(u)$ (this neighbour is a vertex of the forbidden colour gadget H_i that was attached to u'). Furthermore, for all $u' \in U$, since G' is locally irregular, we have that $d_{G'}(u') \neq 2n^3 - i$ for all $i \in \overline{L}(u)$. Equivalently, $d_{G'}(u') = 2n^3 - j$ for any $j \in L(u)$. Thus, $c(u) \in L(u)$ for all $u \in V$. □

Note that the reductions presented in the proofs of Theorems 6.4.5 and 6.4.6 are linear fpt-reductions. Additionally we know that

— there is no algorithm that answers if a graph G of order n has a dominating set of size at most k in time $f(k)n^{o(k)}$ unless the ETH fails [93] and

— there is no algorithm that answers if an instance (G, L) of the LIST COLOURING is a yes-instance in time $\mathcal{O}^*(f(tw)n^{o(tw)})$ unless the ETH fails [61].

So, the following corollary holds.

Corollary 6.4.10. *Let G be a graph of order n and assume the ETH holds. For $k \in \mathbb{N}$, there is no algorithm that decides if $\mathrm{I}(G) \leq k$ in time $f(k)n^{o(k)}$. Furthermore, assuming that G has treewidth tw, there is no algorithm that computes $\mathrm{I}(G)$ in time $\mathcal{O}^*(f(tw)n^{o(tw)})$.*

6.5 Conclusion

In this chapter we introduced the problem of finding a largest locally irregular induced subgraph of a given graph.

There are many interesting directions that could be followed for further research. An obvious one is to investigate whether the problem of calculating $\mathrm{I}(G)$ remains NP-hard for other restricted families of graphs. The first candidate for such a family would be the one of cubic bipartite graphs, through a reduction similar to the one presented in the proof of Theorem 6.2.4 and by carefully adapting the gadget illustrated in Figure 6.1. On the other hand, there are some interesting families, for which the problem of computing an optimal irregulator could be decided in polynomial time, such as chordal graphs. Also, it could be feasible to conceive approximation algorithms for regular bipartite graphs, which have a better approximation ratio than the (simple) algorithm we presented at the end of Section 6.3. The last aspect we find intriguing is to study the parameterised complexity of calculating $\mathrm{I}(G)$ when considering other parameters, like the size of the minimum vertex cover of G, or the order of the remaining graph once the vertices of an irregulator have been removed, with the goal of identifying a parameter that suffices, by itself, in order to have an FPT algorithm.

Another interesting direction would be to study a variation of the problem introduced here, in which the requirement for the desired subgraph to be induced would be dropped. That is, given a graph G, find the minimum number of vertices and/or edges that must be deleted from G in order for the remaining graph to be locally irregular. It could also be interesting to compare the behaviour of such a problem with the problem introduced in this chapter, as the two problems could behave quite differently. As a simple example, consider the path P_5. We know from Theorem 6.2.2 that $\mathrm{I}(P_5) = 1$, and it is easy to check that deleting only one edge of P_5 is not enough for the remaining graph to be locally irregular. Inversely, deleting only the edge v_3v_4 of $P_6 = (v_1, v_2, v_3, v_4, v_5, v_6)$ results in a locally irregular graph, but $\mathrm{I}(P_6) = 2$ (again by Theorem 6.2.2).

Largest connected subgraph games

CHAPTER 7

Preliminaries, first results and hardness

This chapter introduces the largest connected subgraph game and its Maker-Breaker variation. We then proceed by presenting our results when these two games are played on general graphs. Our results for particular classes of graphs, as well as the study of the more particular behaviour of the two games, are presented in the following two chapters.
Concerning the first game introduced here, we first prove that, if Alice plays optimally, then Bob can never win. Then, we define a large class of graphs (called reflection graphs) in which the game is a draw. Then, we show that recognising reflection graphs is GI-hard. We then show that determining the outcome of either game is PSPACE*-complete. The largest connected subgraph game is a joint work with J. Bensmail, F. Mc Inerney and N. Nisse, published in [25] and presented in [24]. A report about the Maker-Breaker version, which is a joint work with J. Bensmail, F. Mc Inerney, N. Nisse and N. Oijid, can be found in [27].*

In the following three chapters we present our work on the newly introduced largest connected subgraph game, as well as its Maker-Breaker variation. Since these two games sometimes behave in a similar fashion, and understanding the behavior of one provides an understanding for the other, we chose to present our work on both of them simultaneously.

We begin this first chapter with the formal definition of both games in Section 7.1. Then, in Section 7.2, we show that Bob can never win the largest connected subgraph game, assuming that Alice plays optimally. Nevertheless, Bob can always guarantee a draw when that game is played on a graph belonging to a class that we call *reflection graphs*. We also show that recognising if a graph is a reflection graph is GI-hard. Finally, in Section 7.3, we deal with the computational complexity of both games. In particular, we show that deciding the outcome of either game is PSPACE-complete, even for restricted families of graphs.

7.1 Preliminaries

Recall that the largest connected subgraph game is played by two players, Alice and Bob, on the same, initially uncoloured graph G. During each turn of the game, each player, starting with Alice, colours an uncoloured vertex with their respective colour; Alice colours vertices red, and Bob blue. The game ends when all vertices of G have been coloured, with the winner being the player whose colour induces the largest connected subgraph of G. If Alice (Bob, resp.) has a winning strategy in the largest connected subgraph game, then G is *A-win* (*B-win*, resp.). If neither Alice nor Bob has a winning strategy in the largest connected subgraph game on G, *i.e.*, the game is a draw if both players follow optimal strategies, then G is *AB-draw*.

Recall that, in general, a Maker-Breaker game played on a graph $G = (V, E)$, with a set of hyperedges $\mathcal{F}$, called the winning sets, being a collection of subsets of V, is a game in which the two players alternatively pick vertices from V. The goal of the first player is to pick vertices forming one set in $\mathcal{F}$, and the goal of the second player is to prevent the first player from doing that. The Maker-Breaker largest connected subgraph game is a Maker-Breaker game played on a graph G. Additionally, there is a positive integer k that is given in the input. The collection $\mathcal{F}$ consists of all the connected subgraphs of G of order at least k. So, in each turn, each player (starting with Alice) colours with their respective colour (Alice with red, Bob with blue) an element of $V(G)$. Alice is considered as the winner if by the end of the game, the subgraph of G induced by the red vertices contains at least one connected subgraph of G of order at least k. Otherwise, Bob wins the game. For a given graph G, we are interested in the parameter $c_g(G)$, which is the largest integer k such that Maker (Alice) has a winning strategy in the Maker-Breaker largest connected subgraph game in G.

In what follows (including the next chapters), we will be using the terms "Scoring game" and "Maker-Breaker game" to refer to the largest connected subgraph game and the Maker-Breaker largest connected subgraph game respectively. It is worth noting here that the former game could, intuitively, be considered as a Maker-Maker game. Nevertheless, formally stating the largest connected subgraph game as a Maker-Maker game is not trivial, as it would require to define the winning sets of the game before the game starts.

The Scoring game is novel in that it is a very natural game which, despite the rich background on these types of games, was not considered before. In particular, it is a connection game (see [42] for more on these games) since the players strive to create connected structures, and it is a *scoring* game since the winner is determined by the scores of the players. When we refer to the *score* of Alice (Bob, resp.) in the context of this game, we mean the largest connected red (blue resp.) component in the graph at the end of the game. The same definition applies for the score of Alice in the context of the Maker-Breaker game, although in this game the score of Bob is not defined. Thus, for a given graph G, $c_g(G)$ is the maximum score for which Alice has a strategy ensuring at least this score in G.

In Figure 7.1 we illustrate an example for each game when played on P_6:

— For the Scoring game: Alice wins if the game is played out as depicted in Figure 7.1 (a), with a score of 2, while Bob achieves a score of 1. However, if the game is played out as depicted in Figure 7.1 (b), then both players achieve the same score of 3, and the game ends in a draw.

— For the Maker-Breaker game: recall that a parameter k is given at the start of the game, and Alice wins only if she achieves a score of at least k. If the game is played out as depicted

(a) One game on P_6 (b) Another game on P_6

Figure 7.1 – An example of playing the Maker-Breaker and the Scoring games on P_6. The colour of each vertex corresponds to the player who chose the vertex, with red (blue resp.) vertices being chosen by Alice (Bob resp.). The white numbers inside the vertices correspond to the turn during which the corresponding vertex was coloured.

in Figure 7.1 (a), then Alice wins if $k \leq 2$, and loses otherwise. If the game is played out as depicted in Figure 7.1 (b), Alice wins if $k \leq 3$ and loses otherwise.

The final definition needed is that of a *strategy*. For both games treated here, a strategy for a player P is a function $\mathcal{S}$ taking all the previous moves of both players (and the order of these moves, hence, the history of the game) as an input, and outputting the next move for player P. Given a graph G, an *optimal strategy for Alice* is a winning strategy when dealing with the Scoring game, and a strategy that ensures her a score of at least $c_g(G)$ when dealing with the Maker-Breaker game. Similarly, an *optimal strategy for Bob* is a strategy that forces the Scoring game to end in a draw, and one that ensures Alice's score is at most $c_g(G)$ when dealing with the Maker-Breaker game. Observe that both games are *parity games* [85]. Indeed, both games can be described as games on a graph, where the players alternatively visit (put a colour) a vertex they desire, provided no player had previously visited that vertex (the vertex was uncoloured), and the outcome depends on the number of vertices each player visited (the number of vertices coloured with their respective colour). Thus, optimal strategies for both games can actually be determined from just the current configuration of coloured vertices, rather than also knowing the order these vertices were coloured in. Thus, for both games, there can also be an equivalent (in terms of optimality) second definition of a strategy for a player P, which is a function $\mathcal{S}$ that takes the current configuration of coloured vertices and outputs the next move for player P. We will interchangeably use both definitions, depending on which one suits us best at the time.

Throughout the next chapters, several of our proofs rely on the fact that Alice or Bob can reach a certain game configuration (*i.e.*, have a certain set of vertices coloured with their colour) early on. In such cases, to lighten the exposition, we will sometimes allow ourselves to expose only the most important moves of the strategies that Alice or Bob must make in some rounds of the game. In particular, the reader should keep in mind that, in each of the strategies we describe, 1) if Alice or Bob cannot colour a given vertex in a given round because that vertex is already coloured, then they must colour any other uncoloured vertex instead, and 2) if no vertex to colour for Alice or Bob in a given round is specified, then they must colour any uncoloured vertex.

We close this introductory section with a general result, concerning both games. As will be seen later on, Alice can exploit different types of strategies to achieve the best possible score for her. One such strategy, which applies to both games and is particularly relevant in sufficiently dense graphs, is through colouring the vertices of a connected dominating set*.

Lemma 7.1.1. *For a graph G, if, at any point in the scoring or the Maker-Breaker game, Alice has coloured all the vertices of a connected dominating set of G, then her score (for either game) will be $\left\lceil \frac{|V(G)|}{2} \right\rceil$.*

$*$. Recall that a connected dominating set of a graph $G = (V, E)$ is a set $S \subseteq V$ such that for every $v \in V$, either $v \in S$ or $v \in N(S)$, and $G[S]$ is connected.

Proof. Assume Alice has coloured the vertices of a connected dominating set S at some point in any of the two games. By the connectivity property of S, there must be, once the game ends, a connected red component containing the vertices of S. Also, by the dominating property of S, all the vertices of G not in S have at least one neighbour in S. This implies that the red subgraph must be connected, and thus, that Alice achieves a score of $\left\lceil \frac{|V(G)|}{2} \right\rceil$. □

7.2 Possible outcomes for the Scoring game, and reflection graphs

In this section we focus on the Scoring game. We begin by showing that there exists no graph that is B-win. The crucial observation here is that in this game, it can never harm a player to have an extra turn, in the sense that an extra turn can never decrease their potential score, *i.e.*, it can never decrease the potential order of the largest connected monochromatic subgraph they can build. This is due to the fact that colouring a vertex only impedes that vertex from being coloured in the future, but does not impede any other vertex from being coloured. Through the use of the classical strategy stealing argument, we can show the following theorem.

Theorem 7.2.1. *There does not exist a graph G that is B-win.*

Proof. Towards a contradiction, assume there exists a graph G that is B-win. Consider the following strategy for Alice. In the first round, Alice colours any arbitrary vertex $v \in V(G)$. Now, one vertex is coloured and it can be assumed that Bob is the first player. Alice now plays according to the second player's winning strategy in G. If, by this strategy, Alice is ever required to colour an already-coloured vertex, then that vertex must be red, and again, in this case, Alice colours any arbitrary uncoloured vertex. Since the only reason a vertex cannot be coloured is that it is already coloured, Alice can always follow this strategy, which is a winning strategy, a contradiction. □

The next natural question to ask is whether there exist graphs that are A-win (AB-draw, resp.). It is easy to see that there are an infinite number of A-win graphs as any star (of order not equal to 2) is A-win, since, in order to win, it is sufficient for Alice to colour the universal vertex in the first round. This also illustrates that there are an infinite number of A-win graphs for which, through optimal strategies, the order of the largest connected red subgraph is arbitrarily bigger than the order of the largest connected blue subgraph. There are also an infinite number of AB-draw graphs, since any graph of even order with two universal vertices is clearly AB-draw (as, to ensure at least a draw, it is sufficient for Bob to colour a universal vertex in the first round). By adding an isolated vertex to any of the graphs mentioned in the previous sentence, we also have that there are an infinite number of AB-draw graphs of odd order. In Section 8.1, we will see that any path of order at least 11 is AB-draw, and hence, that there exists an infinite family of connected graphs of odd order that are AB-draw.

We can actually define a much richer class of graphs that are AB-draw.

Definition 7.2.2. *A reflection graph is any graph G, whose vertices can be partitioned into two sets $U = \{u_1, \dots, u_n\}$ and $V = \{v_1, \dots, v_n\}$ such that:*

1. *the subgraph $G[U]$ induced by the vertices of U is isomorphic to the subgraph $G[V]$ induced by the vertices of V, and the function mapping u_i to v_i for all $1 \leq i \leq n$, is an isomorphism between $G[U]$ and $G[V]$;*
2. *for any two vertices $u_i \in U$ and $v_j \in V$, if the edge u_iv_j exists, then the edge u_jv_i exists (where, for any $1 \leq \ell \leq n$, $v_\ell \in V$ is the image of $u_\ell \in U$ by the said isomorphism).*

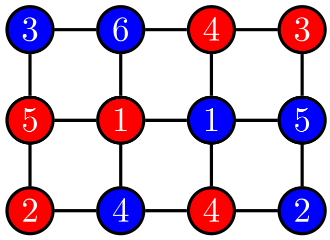


Figure 7.2 – An example of the Scoring game on a reflection graph. Both players achieve the same score, equal to 3.

In other words, if a graph G can be formed by taking two copies of a graph H, and adding edges between both copies of H according to the second condition above, then G is a reflection graph. It is easy to see, for example, that paths and cycles of even order, and Cartesian grids of even order, are reflection graphs. In Figure 7.2 we illustrate an example of playing the Scoring game on a Cartesian grid of order 12.

Proposition 7.2.3. *Paths, cycles, Cartesian grids, and king grids of even order are reflection graphs.*

Proof. Note that a path of even order $2n$ ($n \geq 1$) can be regarded as the disjoint union of two paths $(u_1, \dots, u_n)$ and $(v_1, \dots, v_n)$ of order n, connected by the edge u_1v_1. Similarly, a cycle of even length $2n$ ($n \geq 2$) can be regarded as the disjoint union of two paths $(u_1, \dots, u_n)$ and $(v_1, \dots, v_n)$ of order n, joined by the edges u_1v_1 and u_nv_n. Thus, paths and cycles of even order are reflection graphs. Now, for a Cartesian grid H to be of even order, at least one of its two dimensions must be even. Assume, w.l.o.g., that its number of columns $2n$ ($n \geq 1$) is even, while its number of rows is $m \geq 1$. Then, H can be regarded as the disjoint union of two Cartesian grids G_1 and G_2 with m rows and n columns each, being joined by the edges $u_1v_1, \dots, u_mv_m$ if we denote by $u_1, \dots, u_m$ the consecutive vertices of the last column of G_1, and by $v_1, \dots, v_m$ the consecutive vertices of the first column of G_2. King grids of even order can similarly be described that way (note that we also have both edges u_iv_{i+1} and $u_{i+1}v_i$ for every $i \in \{1, \dots, m-1\}$). Thus, Cartesian grids and king grids of even order are reflection graphs. □

The next theorem proves that reflection graphs are AB-draw.

Theorem 7.2.4. *Any reflection graph G is AB-draw.*

Proof. We define a "copying" strategy for Bob which guarantees a draw. Let $U = \{u_1, \dots, u_n\}$ and $V = \{v_1, \dots, v_n\}$ be a partitioning of the vertices of G that satisfies the two conditions required for G to be a reflection graph. Bob's copying strategy is as follows. In every round, when Alice colours a vertex $u_i \in U$ ($v_i \in V$, resp.), Bob colours its image $v_i \in V$ ($u_i \in U$, resp.). By Bob's strategy, it is easy to see that Bob can always play in this way. Moreover, by the symmetry of the graph, for every vertex coloured red (blue, resp.) in U, its image is coloured blue (red, resp.) in V. Hence, once all vertices are coloured, by the symmetry of the graph and the second condition for reflection graphs concerning the edges between vertices of U and V, there is a blue isomorphic copy of any connected red subgraph in G. Thus, the game ends in a draw. □

It turns out that recognising reflection graphs is not an easy problem. We show that it is GI-hard, meaning that it is at least as hard as the GRAPH ISOMORPHISM problem [86]. This essentially shows that recognising reflection graphs is unlikely to be polynomial-time solvable as there exist problems in GI (notably the GRAPH ISOMORPHISM problem) which are good candidates for being NP-intermediate, *i.e.*, in the class NPI, which is the complexity class of problems that are in NP but that are neither NP-hard nor in P. Note that the class NPI is non-empty if and only if P$\neq$ NP.

Theorem 7.2.5. *Given a graph G, deciding if G is a reflection graph is GI-hard.*

Proof. The reduction is from the GRAPH ISOMORPHISM problem, in which, given two input graphs G_1 and G_2, one has to decide whether G_1 and G_2 are isomorphic. We may further assume that G_1 and G_2 are each connected and of odd order, which is one of the input restrictions for which the problem remains hard. Indeed, note that we obtain an equivalent instance of the problem (with the desired properties), upon adding, if needed, one or two universal vertices to both G_1 and G_2.

We construct a graph H in polynomial time, such that G_1 and G_2 are isomorphic if and only if H is a reflection graph. The graph H we construct is simply $G_1 + G_2$, the disjoint union of G_1 and G_2. Let us prove the two directions of the equivalence.

First, we prove the forward direction. Assume that the vertices of G_1 and G_2 are $u_1, \dots, u_n$ and $v_1, \dots, v_n$, respectively, ordered in such a way that there is an isomorphism between G_1 and G_2 where v_i is the image of u_i, for all $1 \leq i \leq n$. Note that no edge joins a vertex from G_1 and a vertex from G_2. Then, $G_1 \cup G_2 = H$ is a reflection graph with $U = V(G_1)$ and $V = V(G_2)$. The reflection property is trivial in that case.

Now, we prove the other direction. Assume that H is a reflection graph with parts $U = \{u_1, \dots, u_n\}$ and $V = \{v_1, \dots, v_n\}$ such that the function mapping u_i to v_i (for all $1 \leq i \leq n$) is an isomorphism between $H[U]$ and $H[V]$. If U is precisely $V(G_1)$ while V is precisely $V(G_2)$, then we get that $H[U] = G_1$ and $H[V] = G_2$ are isomorphic, by the definition of a reflection graph. So, assume this is not the case.

For all $1 \leq i \leq n$, note that either

1) $u_i \in V(G_1)$ and $v_i \in V(G_2)$,
2) $u_i \in V(G_2)$ and $v_i \in V(G_1)$,
3) $u_i, v_i \in V(G_1)$, or
4) $u_i, v_i \in V(G_2)$.

We consider all i's in turn, and possibly switch vertices of U and V as follows:

— If u_i and v_i satisfy Condition 1) above, then we do nothing.

— If u_i and v_i satisfy Condition 2) above, then we move u_i from U to V, and, conversely, move v_i from V to U, resulting in a partition of $V(H)$ into two equally sized parts U' and V'. Note that, considering the ordering $u'_1, \dots, u'_n$ and $v'_1, \dots, v'_n$ of U' and V' (where $u'_j = u_j$ and $v'_j = v_j$ for all $1 \leq j \leq n$ such that $i \neq j$, and $u'_i = v_i$ and $v'_i = u_i$), respectively, we have that H is also a reflection graph with respect to the two parts U' and V'. Indeed, by the isomorphism and reflection properties, we have that u_i was neighbouring $u_{i_1}, \dots, u_{i_k}$ in U (and so, v_i was neighbouring $v_{i_1}, \dots, v_{i_k}$ in V) and $v_{j_1}, \dots, v_{j_k}$ in V (and so, v_i was neighbouring $u_{j_1}, \dots, u_{j_k}$ in U), which translates, for U' and V', into u'_i neighbouring $v_{j_1}, \dots, v_{j_k}$ in V' (and so, v'_i neighbouring $u_{j_1}, \dots, u_{j_k}$ in U') and $u_{i_1}, \dots, u_{i_k}$ in U' (and so, v'_i neighbouring $v_{i_1}, \dots, v_{i_k}$ in V').

— If u_i and v_i satisfy Condition 3) or 4) above, then we get a contradiction to one of the original assumptions on G_1 and G_2. Indeed, assume, w.l.o.g., that u_i and v_i satisfy Condition 3), *i.e.*, both u_i and v_i originate from G_1. Note that, because G_1 and G_2 are each connected and of odd order, there must be a pair u_j, v_j such that, w.l.o.g., $u_j \in V(G_1)$ and $v_j \in V(G_2)$. Furthermore, since G_1 is connected, for such a pair u_j, v_j, it can be assumed, w.l.o.g., that at least one of u_iu_j and v_iu_j is an edge. If the former edge exists, then the contradiction arises from the fact that, since H is a reflection graph, we must have the edge v_iv_j as well, which is not possible since $v_i \in V(G_1)$ and $v_j \in V(G_2)$. If the latter edge exists, then, because H is a reflection graph, the edge u_iv_j also exists, hence, an edge between G_1 and G_2, which again is a contradiction.

Once all i's have been treated this way, H remains a reflection graph, and a direct isomorphism between G_1 and G_2 is deduced. □

7.3 Both games are PSPACE-hard

In this section we show that both the Scoring game and the Maker-Breaker are PSPACE-complete. For the former, this means that given a graph G, deciding if it is A-win or AB-draw is PSPACE-complete. For the latter game, given a graph G and an integer $k \geq 1$, deciding whether $c_g(G) \geq k$ is PSPACE-complete.

Before we start, let us argue about the fact that the decision problems defined above for both games belong in PSPACE. This follows from the fact that for both games, there are $\lceil |V(G)|/2 \rceil$ rounds and the number of possible moves for each player in a round is at most $|V(G)|$. Thus, in the upcoming proofs, we focus on proving the PSPACE-hardness of these games.

Note that the proof of PSPACE-hardness for the Scoring game is more involved than the corresponding proofs for the Maker-Breaker game on bipartite and split graphs, although they share many similarities. Thus, we chose to present our reductions starting with two of our results on the complexity of the Maker-Breaker game, as to render the proof for the Scoring game easier to follow.

7.3.1 The Maker-Breaker game on planar graphs

In this section we prove that the Maker-Breaker game remains PSPACE-complete even when the graphs on which the game is played is assumed planar. This result is proven by a reduction from PLANAR GENERALISED HEX, which was proven to be PSPACE-complete [109]. PLANAR GENERALISED HEX is played on a planar graph G, in which a particular *outside pair* $\{s, t\}$ of vertices, *i.e.*, $st \notin E(G)$ and $G + st$ is planar, is set. Initially, s and t are red. Then, in successive rounds, the first player, Alice, colours an uncoloured vertex red, before the second player, Bob, then colours an uncoloured vertex blue. The game ends once all the vertices of G have been coloured. If the red subgraph contains a path joining s and t, then Alice wins. Otherwise, Bob wins.

Theorem 7.3.1. *Given a graph G and an integer $k \geq 1$, it is PSPACE-complete to decide whether $c_g(G) \geq k$, even when G is restricted to be in the class of planar graphs.*

Proof. The reduction is from PLANAR GENERALISED HEX. Let (H, s, t) be an instance of PLANAR GENERALISED HEX such that H is the planar graph with the outside pair $\{s, t\}$, that the

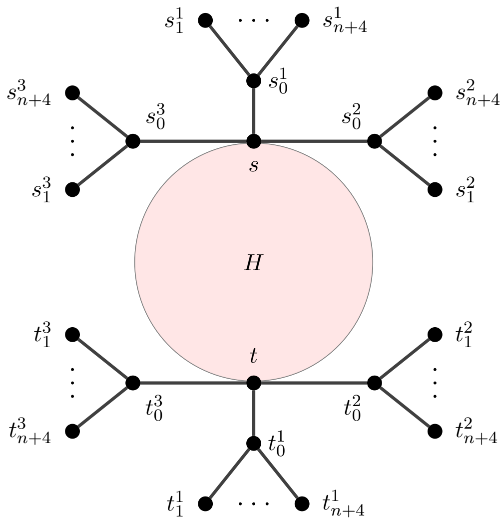


Figure 7.3 – Illustration of the construction in the proof of Theorem 7.3.1.

game is being played on. Set $n = |V(H)|$. By adding a degree-1 vertex (a leaf) in H if needed, we can suppose n is even, as this will not change the outcome of (H, s, t). Let G be the graph constructed as follows (see Figure 7.3). Start from G being the graph H. Then, add three vertices s_0^1, s_0^2, s_0^3 and make each of them adjacent to s, and add another three vertices t_0^1, t_0^2, t_0^3, and make each of those adjacent to t. Finally, to each of these six vertices we have just added, attach $n + 4$ new degree-1 vertices, so that a total of $6(n + 4)$ degree-1 vertices (leaves) are added to G. The construction is achieved in polynomial time, and since H is planar, G is too.

Set $k = n + 5$. We will show that Alice wins in (H, s, t) if and only if $c_g(G) \geq k$. Let us assume first that Alice has a winning strategy in (H, s, t). We give a strategy for Alice that ensures a score of at least k when playing the Maker-Breaker Largest Connected Subgraph game in G. In the first round, Alice colours s. In the second round, Alice colours s_0^1 if possible, and if not, then she colours s_0^2. From the third round on,

— if Alice can colour a vertex in $\{s_0^1, s_0^2, s_0^3\}$ in the third round, then she does. If so, then, in each of the next rounds, if possible, Alice colours an uncoloured neighbour of an s_0^i she coloured earlier. At the end of the game, the red subgraph will contain a connected component of order at least $3 + \left\lceil \frac{2(n+4)-3}{2} \right\rceil = n + 6$, and thus, Alice will have a score of at least k;

— otherwise, Bob coloured two vertices in $\{s_0^1, s_0^2, s_0^3\}$ in the first two rounds. Then, Alice colours t in the third round, and she then colours one of t_0^1 and t_0^2 in the fourth round. At this point, for the same reasons as earlier, if Bob has not coloured two vertices in $\{t_0^1, t_0^2, t_0^3\}$ by the end of the fourth round, then Alice can colour a vertex in that set in the fifth round, and, as above, guarantee herself a score of at least k.

Thus, we can suppose that, after four rounds, w.l.o.g., s, t, s_0^1, and t_0^1 are red, while s_0^2, s_0^3, t_0^2, and t_0^3 are blue. From here, Alice's strategy continues as follows. In the fifth round, Alice colours, in G, the vertex of H she would have coloured in the first round of her winning strategy in (H, s, t). From the sixth round on, in each round, if the last vertex coloured by Bob in G is

— some vertex $u \in V(H)$, then Alice colours, in G, the vertex of H she would have coloured in her winning strategy in (H, s, t), as an answer to Bob colouring u;

— a leaf adjacent to some s_0^i or t_0^i, then Alice colours another uncoloured leaf adjacent to the same vertex.

Whenever Alice cannot follow the strategy above, she colours any arbitrary vertex. By this strategy, at the end of the game in G, s and t are red, and all the vertices that Alice would have coloured through her winning strategy in (H, s, t) are also red. Moreover, s_0^1 and t_0^1 are red, and, for each of them, she coloured half of their adjacent leaves. Thus, the red subgraph contains a connected component of order at least $n + 8$. Thus, Alice achieves a score of at least k, and $c_g(G) \geq k$.

Assume now that Bob has a winning strategy in (H, s, t). We give a strategy for Bob that ensures that Alice's score is strictly less than k when playing the Maker-Breaker Largest Connected Subgraph game in G. In each round, if the last vertex coloured by Alice is

— in $\{s, s_0^1, s_0^2, s_0^3\}$, then Bob colours a vertex in $\{s, s_0^1, s_0^2, s_0^3\}$;

— in $\{t, t_0^1, t_0^2, t_0^3\}$, then Bob colours a vertex in $\{t, t_0^1, t_0^2, t_0^3\}$;

— a vertex u of $H - \{s, t\}$, then Bob colours the vertex of H he would have coloured by his winning strategy in (H, s, t), as an answer to Alice colouring u;

— a leaf adjacent to some s_0^i or t_0^i, then Bob colours another uncoloured leaf adjacent to the same vertex.

Note that Bob always answers to one of Alice's moves by colouring a vertex in a set with even size since n is even. Thus, Bob can follow this strategy from start to end. At the end of the game in G, the largest connected component of the red subgraph cannot contain both s and t, as the moves made by Alice and Bob correspond exactly to the moves that would have been made if they had played in (H, s, t). Moreover, there cannot be two s_0^i's belonging to the same connected red component, as, by the strategy above, Bob must have coloured s in this case. The same holds for the t_0^i's. Also, for any of the s_0^i's and t_0^i's, by Bob's strategy above, Alice can have coloured at most half of the leaves adjacent to it. Thus, because Alice coloured at most half of the vertices in $H - \{s, t\}$, the largest connected red component in G must have order at most $\frac{n-2}{2} + 2 + \frac{n+4}{2} = n + 3$. Thus, Alice achieves a score of less than k, and $c_g(G) < k$. $\square$

7.3.2 The Maker-Breaker game on bipartite and split graphs

The PSPACE-completeness results presented in the current and the following section, are established via reductions from POS CNF, a game for which deciding whether Alice or Bob has a winning strategy was shown to be PSPACE-complete in [111].

Definition 7.3.2 (POS CNF)**.** *A 2-player game, where the input consists of a set of variables $X = \{x_1, \dots, x_n\}$ and a conjunctive normal form (thus the term CNF) formula ϕ consisting of clauses $C_1, \dots, C_m$ that each contain only variables from X, all of which appear in their positive form (thus the term POS). In each round, the first player, Alice, first sets a variable (that is not yet set) to true, and then, the second player, Bob, sets a variable (that is not yet set) to false. Once all the variables have been assigned a truth value, Alice wins if ϕ is true, and Bob wins if ϕ is false.*

It is worth noting here that both of the reductions we provide from POS CNF (the ones presented in this and the following section) share a lot of similarities to the one given in [62]. The game studied in [62] is the Maker-Breaker game played on a graph G, with players picking vertices of G, and the winning sets consists of all the dominating sets of G. Taking into account Lemma 7.1.1 and the fact that such reductions are rather standard, it is not surprising that the following proofs are similar to the corresponding one in [62]. Nevertheless, we include the full proofs for completeness.

Theorem 7.3.3. *Given a graph G and an integer $k \geq 1$, it is* PSPACE*-complete to decide whether $c_g(G) \geq k$, even when G is restricted to be in the class of bipartite graphs of diameter* 4.

Proof. We prove the PSPACE-hardness via a reduction from POS CNF. Let (X, ϕ) be an instance of POS CNF. Set $X = \{x_1, \dots, x_n\}$ and $\phi = C_1 \wedge \dots \wedge C_m$. By adding a dummy variable in X if needed, we can suppose n is even.

Consider the graph G constructed as follows. For every variable $x_i \in X$, we add a vertex v_i to G. For every clause C_j of ϕ, we add two vertices C_j^1 and C_j^2 to G. For every variable $x_i \in X$ and clause C_j of ϕ, we add the edges $v_iC_j^1$ and $v_iC_j^2$ to G if x_i appears in C_j. Finally, we add two vertices u_1 and u_2 to G, that we make adjacent to all of the v_i's. Note that the resulting G, which is constructed in polynomial time, is bipartite and has diameter at most 4.

Set $k = |V(G)|/2$, and note that $|V(G)|$ is even. We will show that Alice wins in (X, ϕ) if and only if $c_g(G) \geq k$. Let us assume first that Alice has a winning strategy in (X, ϕ). We give a strategy for Alice that ensures a score of at least k when playing the Maker-Breaker Largest Connected Subgraph game in G. In the first round, Alice colours the vertex v_i that corresponds to the variable $x_i \in X$ she would have set to true in the first round of her winning strategy in (X, ϕ). From the second round on, in each round, if the last vertex coloured by Bob is

- some v_i, then Alice colours the vertex v_j corresponding to the variable x_j she would set to true in response to Bob setting x_i to false in her winning strategy in (X, ϕ);
- u_1 (u_2, resp.), then Alice colours u_2 (u_1, resp.);
- some C_j^1 (C_j^2, resp.), then Alice colours C_j^2 (C_j^1, resp.).

Whenever Alice cannot follow the strategy above, she colours any arbitrary vertex. By Alice's strategy, once the game in G ends, exactly one vertex in every pair $\{C_j^1, C_j^2\}$ is red, exactly one vertex in $\{u_1, u_2\}$ is red, and the v_i's corresponding to the x_i's she would have set to true in her winning strategy for (X, ϕ) are also red. Because Alice wins in (X, ϕ) with that strategy, every vertex C_j^ℓ of G coloured red must be adjacent to at least one vertex v_k coloured red corresponding to a variable she would have set to true when playing in (X, ϕ). Since all the v_i's are dominated by u_1 and u_2, and one of these two vertices is red, we deduce that the red subgraph must contain only one connected component. Thus, Alice achieves a score of k and $c_g(G) \geq k$.

Assume now that Bob has a winning strategy in (X, ϕ). We give a strategy for Bob that ensures that Alice's score is strictly less than k when playing the Maker-Breaker Largest Connected Subgraph game in G. In each round, if the last vertex coloured by Alice is

- some v_i, then Bob colours the vertex v_j corresponding to the variable x_j he would set to false in response to Alice setting x_i to true in his winning strategy in (X, ϕ);
- u_1 (u_2, resp.), then Bob colours u_2 (u_1, resp.);
- some C_j^1 (C_j^2, resp.), then Bob colours C_j^2 (C_j^1, resp.).

Note that Bob can follow this strategy from start to end, as n is even. By Bob's strategy, once the game in G ends, exactly one vertex in every pair $\{C_j^1, C_j^2\}$ is red. Also, since Bob coloured all the v_i's corresponding to x_i's he would set to false when following a winning strategy in (X, ϕ), there exists a C_q that is not satisfied in (X, ϕ), meaning its variables were all set to false by Bob. In G, this translates to exactly one of C_q^1 or C_q^2 being red while all of their neighbours (the v_i's corresponding to the x_i's that C_q contains), are blue. Thus, the red subgraph contains at least two connected components, and hence, Alice achieves a score of less than k, and $c_g(G) < k$. □

Corollary 7.3.4. *Given a graph G and an integer $k \geq 1$, it is PSPACE-complete to decide whether $c_g(G) \geq k$, even when G is restricted to be in the class of split graphs.*

Proof. The proof is similar to that of Theorem 7.3.3, with the slight difference being in the construction of G. Here, neither of the vertices u_1 and u_2 are added, while all the possible edges between the v_i's are added so that they form a clique, thus making G a split graph. The same strategies for Alice and Bob (omitting u_1 and u_2) from the proof of Theorem 7.3.3 remain applicable by the same arguments, and the result follows. □

7.3.3 The Scoring game on bipartite graphs

In this section, we show that the Scoring game is PSPACE-complete, even when restricted to bipartite graphs of small diameter.

Theorem 7.3.5. *Given a graph G, deciding if G is A-win is PSPACE-complete, even if G is bipartite and has a diameter of 5.*

Proof. Since the number of rounds is exactly $\lceil |V(G)|/2 \rceil$ and there are at most $|V(G)|$ possible moves for a player in any round, the decision problem is in PSPACE. To prove the problem is PSPACE-hard, we give a reduction from POS CNF. By adding a dummy variable (if necessary), it is easy to see that POS CNF remains PSPACE-hard even if the number of variables n is odd. From an instance ϕ of POS CNF where n is odd, we construct, in polynomial time, an instance G of the largest connected subgraph game such that Alice wins in ϕ if and only if G is A-win. Let $x_1, \dots, x_n$ be the variables and let $C_1, \dots, C_m$ be the clauses of ϕ. The construction of G is as follows (see Figure 7.4 for an illustration): for each variable x_i ($1 \leq i \leq n$), there is a vertex x_i, and, for each clause C_j ($1 \leq j \leq m$), there are 6 vertices $C_j^1, \dots, C_j^6$. For all $1 \leq i \leq n$ and $1 \leq j \leq m$, if the variable x_i appears in the clause C_j, then there is the edge $x_i C_j^q$ for all $1 \leq q \leq 6$. In addition to this, there are the vertices u, v_1, v_2, w_1, w_2, and $y_1, \dots, y_{n+6m-2}$, and the edges w_1v_1, v_1u, uv_2, and v_2w_2. Furthermore, for all $1 \leq i \leq n$, there is the edge ux_i, and, for all $1 \leq \ell \leq n + 6m - 2$, there are the edges $w_1 y_\ell$ and $w_2 y_\ell$. This completes the construction. To simplify the proof, let P be the subgraph of G induced by the vertices x_i ($1 \leq i \leq n$) and C_j^q ($1 \leq q \leq 6$ and $1 \leq j \leq m$), and let Q be the subgraph of G induced by the vertices in $V(G) \setminus (V(P) \cup \{u\})$. Note that u separates P from Q. To simplify the upcoming calculations, let $b = (n-1)/2 + 3m + 1 = \lfloor n/2 \rfloor + 3m + 1$ since n is odd.

We start by proving the first direction, that is, if Alice wins in ϕ, then G is A-win. We describe a winning strategy for Alice. In what follows, whenever Alice cannot follow her strategy, she simply colours any arbitrary vertex and resumes her strategy for the subsequent moves of Bob. Alice first colours u. Now, Bob can only construct connected blue subgraphs in P or Q since u separates them. For all $1 \leq j \leq m$, whenever Bob colours a vertex in $\{C_j^1, \dots, C_j^6\}$, then Alice also colours a vertex in $\{C_j^1, \dots, C_j^6\}$, so in what follows, we may assume that Bob does not colour such a vertex. There are two cases depending on Bob's next move.

— Bob colours a vertex in Q. Then, Alice colours the vertex x_i that corresponds to the variable x_i she wants to set to true in her winning strategy in ϕ. Now, whenever Bob colours a vertex x_p ($1 \leq p \leq n$ and $p \neq i$), Alice assumes Bob set the variable x_p to false in ϕ and colours the vertex in $\{x_1, \dots, x_n\}$ corresponding to her winning strategy in ϕ. Otherwise, whenever Bob colours a vertex in Q, then Alice colours a vertex in Q. Note that, by this strategy, Alice ensures a connected red subgraph of order at least $\lceil n/2 \rceil + 3m + 1 = b + 1$

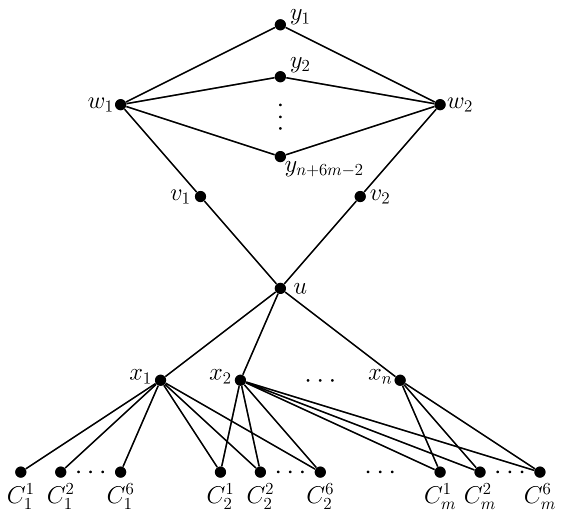


Figure 7.4 – An example of the construction of the graph G in the proof of Theorem 7.3.5, where, among other variables, the clause C_1 contains the variable x_1, the clause C_2 contains the variables x_1 and x_2, and the clause C_m contains the variables x_2 and x_n.

since she colours half the variable vertices (rounded up since Alice starts in these vertices), half the clause vertices, and u, and since she followed a winning strategy in ϕ, this subgraph is indeed connected. Furthermore, she ensures that any connected blue subgraph in P is of order at most $\lfloor n/2 \rfloor + 3m = b - 1$, and hence, Bob must construct his largest connected blue subgraph in Q if he wants to manage a draw.
If Alice colours v_1 or v_2 she wins, since then she ensures a connected red subgraph of order at least $\lceil n/2 \rceil + 3m + 2 = b + 2$, while she ensures that any connected blue subgraph in Q is of order at most $b + 1$. Indeed, $|V(Q)| = n + 6m + 2$, Bob first colours one vertex in Q, and then each subsequent time he colours a vertex in Q, Alice does the same, and thus, Bob colours at most $1 + \lceil (n + 6m + 1)/2 \rceil = 1 + (n + 1)/2 + 3m = b + 1$ vertices in Q. Thus, Bob must have coloured v_1 and v_2 in the first two rounds. Now, Alice colours w_2 (so that v_2 cannot be part of a large connected blue subgraph since u is red), and she wins since she ensures that any connected blue subgraph in Q is of order at most $\lceil (|V(Q)| - 3)/2 \rceil + 1 = \lceil (n + 6m - 1)/2 \rceil + 1 = b$ (recall that n is odd).

— Bob colours a vertex in $\{x_1, \ldots, x_n\}$. Then, Alice colours w_2.
First, let us assume that Bob does not colour v_2 during his second turn,. Then, Alice will colour v_2 in the next round and win with the following strategy: whenever Bob colours a vertex
  — in $\{w_1, v_1\}$, then Alice colours the other vertex in $\{w_1, v_1\}$;
  — y_ℓ ($1 \le \ell \le n + 6m - 2$), then Alice colours a vertex y_k ($1 \le k \le n + 6m - 2$ and $\ell \ne k$);
  — x_i ($1 \le i \le n$), then Alice colours a vertex x_p ($1 \le p \le n$ and $i \ne p$).
In this way, Alice has coloured u, v_2, and w_2. Moreover, she coloured at least half (rounded down) of the $n + 6m - 2 + n - 2$ vertices y_ℓ and x_i that remained uncoloured after Bob's second turn (at most two vertices y_ℓ and x_i could have been coloured blue during Bob's first two turns, and the half is rounded down since Alice plays in second in this set of

vertices). In total, this guarantees a connected red subgraph of order at least $3 + \lfloor (n + 6m - 2 + n - 2)/2 \rfloor = n + 3m + 1 > b + 1$ without counting any of the vertices C_j^q ($1 \leq q \leq 6$, $1 \leq j \leq m$). Regarding Bob, any connected blue subgraph in P has order at most $2 + \lceil (|V(P)| - 2)/2 \rceil$ (since, except for the first two turns of Bob, Alice always answers in P when Bob colours a vertex in P), *i.e.*, at most $2 + \lceil (n + 6m - 2)/2 \rceil = 2 + \lceil n/2 \rceil + 3m - 1 = 1 + (n+1)/2 + 3m = (n-1)/2 + 3m + 2 = b + 1$. Moreover, any connected blue subgraph in Q has order at most $\lceil (|V(Q)| - 3)/2 \rceil + 1 = \lceil (n + 6m - 1)/2 \rceil + 1 = (n-1)/2 + 3m + 1 = b$. Hence, Alice wins in this case.
Second, let us assume that Bob colours v_2 during his second turn. Now, Alice colours w_1 and Bob is forced to colour v_1 for the same reasons as above. Alice now colours y_1 and then she follows the strategy just previously described above (the one for the case where Bob did not colour v_2). In this way, Alice ensures a connected red subgraph containing w_2, w_1, and half of the vertices y_ℓ (rounded up), *i.e.*, a connected red subgraph of order at least $\lceil (n + 6m - 2)/2 \rceil + 2 = \lceil n/2 \rceil + 3m + 1 = b + 1$ in Q. Regarding Bob, any connected blue subgraph in P has at most $\lceil (|V(P)| - 1)/2 \rceil + 1$ vertices (the first vertex that Bob coloured, plus half of the remaining vertices in P), *i.e.*, at most $\lceil (n + 6m - 1)/2 \rceil + 1 = (n - 1)/2 + 3m + 1 = b$ vertices, and any connected blue subgraph in Q has at most one vertex. Hence, Alice wins in this case as well, and this concludes the proof of the first direction.

Now, we prove the other direction, that is, if Bob wins in ϕ, then G is AB-draw. We give a strategy for Bob that guarantees the game in G is a draw. In what follows, whenever Bob cannot follow his strategy, he simply colours any arbitrary vertex and resumes his strategy for the subsequent moves of Alice. Part of Bob's strategy is as follows: whenever Alice colours

— a vertex in $\{C_j^1, \ldots, C_j^6\}$ for some $1 \leq j \leq m$, then Bob also colours a vertex in $\{C_j^1, \ldots, C_j^6\}$;
— a vertex x_i for some $1 \leq i \leq n$, then Bob assumes Alice set the variable x_i to true in ϕ and colours the vertex in $\{x_1, \ldots, x_n\}$ corresponding to his winning strategy in ϕ.

Hence, we just need to describe a strategy for Bob in Q', the subgraph of G induced by the vertices in $V(Q) \cup \{u\}$. W.l.o.g., we may assume that the first vertex Alice colours in Q' is neither v_2 nor w_2. Then, Bob colours w_2. Now, if the first two vertices Alice colours in Q' are:

— w_1 and v_1, then Bob colours u. Now, Alice must colour v_2, as otherwise, Bob wins as in the proof of the first direction where Alice wins if she manages to colour w_2, v_2, and u. Then, Bob colours y_k for some $1 \leq k \leq n + 6m - 2$. Now, whenever Alice colours a vertex y_ℓ ($1 \leq \ell \leq n + 6m - 2$), then Bob colours a vertex y_k ($1 \leq k \leq n + 6m - 2$ and $\ell \neq k$);
— w_1 and v_2, then Bob colours y_ℓ for some $1 \leq \ell \leq n + 6m - 2$. Now, whenever Alice colours a vertex in $\{v_1, u\}$, then Bob colours the other vertex in $\{v_1, u\}$. Otherwise, whenever Alice colours a vertex y_ℓ ($1 \leq \ell \leq n + 6m - 2$), then Bob colours a vertex y_k ($1 \leq k \leq n + 6m - 2$ and $\ell \neq k$);
— w_1 and u, then Bob colours v_1. Now, whenever Alice colours a vertex in $\{y_1, \ldots, y_{n+6m-2}, v_2\}$, then Bob colours another vertex in $\{y_1, \ldots, y_{n+6m-2}, v_2\}$;
— w_1 and y_k for some $1 \leq k \leq n + 6m - 2$, then Bob colours v_2. Now, Alice must colour u, as otherwise, Bob wins as in the proof of the first direction where Alice wins if she manages to colour w_2, v_2, and u. Then, Bob colours v_1. Now, whenever Alice colours a vertex y_ℓ ($1 \leq \ell \leq n + 6m - 2$), then Bob colours a vertex y_p ($1 \leq p \leq n + 6m - 2$ and $\ell \neq p$);

— any other combination, then Bob colours w_1. Now, whenever Alice colours a vertex in $\{y_1, \ldots, y_{n+6m-2}, v_1, v_2, u\}$, then Bob colours a different vertex in $\{y_1, \ldots, y_{n+6m-2}, v_1, v_2\}$ (note that u is not included here).

In the first two cases above, there is a connected blue component in Q (consisting of w_2 and half of the vertices y_ℓ, rounded up since Bob starts in these vertices) of order at least $\lceil (n+6m-2)/2 \rceil + 1 = (n-1)/2 + 3m + 1 = b$. In the third case above, there is a connected blue component in Q of order at least $\lfloor (n+6m-2+1)/2 \rfloor + 1 = (n-1)/2 + 3m + 1 = b$ (consisting of w_2 and half of the vertices y_ℓ and v_2, rounded down since Alice starts in these vertices). In the fourth case above, there is a connected blue component in Q of order at least $\lfloor (n+6m-3)/2 \rfloor + 2 = (n-1)/2 + 3m + 1 = b$ (the two vertices w_2 and v_2, plus half, rounded down, of the vertices y_ℓ minus the first one coloured by Alice). In the last case above, there is a connected blue component in Q of order at least $\lfloor (n+6m-2)/2 \rfloor + 2 = \lfloor n/2 \rfloor + 3m + 1 = (n-1)/2 + 3m + 1 = b$ (indeed, if α, β are the first two vertices in Q' coloured by Alice, then Bob colours at least w_1, w_2, and half of the vertices in $\{y_1, \ldots, y_{n+6m-2}, v_1, v_2\} \setminus \{\alpha, \beta\}$ rounded down). To summarise, in each of the cases, Bob has ensured that there is a connected blue component in Q of order at least b.

Regarding Alice, in the first two cases above, any connected red component in Q is of order at most $\lfloor (n+6m-2)/2 \rfloor + 2 = (n-1)/2 + 3m + 1 = b$ (consisting of at most w_1, v_1, and half of the vertices y_ℓ rounded down). In the third case above, any connected red component in Q is of order at most $\lceil (n+6m-2)/2 \rceil + 1 = \lceil n/2 \rceil + 3m = (n-1)/2 + 3m + 1 = b$ (consisting of at most w_1 and half of the vertices y_ℓ rounded up). In the fourth case above, any connected red component in Q is of order at most $\lceil (n+6m-3)/2 \rceil + 2 = (n-1)/2 + 3m + 1 = b$ (consisting of at most w_1, one vertex y_k, and half of the remaining $n+6m-3$ vertices y_ℓ rounded up). In the last case above, any connected red component in Q is of order at most 1. To summarise, in each of the cases, Bob has ensured that any connected red component in Q is of order at most b. Hence, if Alice is to win, she must have constructed a connected red component of order at least $b+1$ in P', the subgraph of G induced by $V(P) \cup \{u, v_1, v_2\}$ (since, by Bob's strategy, it can never be that u, v_1, and w_1 (u, v_2, and w_2, resp.) are all red). Since Bob follows a winning strategy in ϕ whenever Alice colours a vertex in $\{x_1, \ldots, x_n\}$, there is at least one j ($1 \leq j \leq m$) for which none of the vertices in $C_j^1, \ldots, C_j^6$ are adjacent to a red vertex. Hence, any connected red component in P' is of order at most $\lceil (n+6m-6)/2 \rceil + 3 = \lceil n/2 \rceil + 3m = (n-1)/2 + 3m + 1 = b$. Thus, in G, there is a connected blue component of order at least b and any connected red component is of order at most b. Hence, Alice does not win in any of the cases and this concludes the proof of the second direction. □

7.4 Conclusion

In this chapter we introduced the largest connected subgraph game and its Maker-Breaker variation. Apart from establishing that deciding the outcome of either game is computationally hard in general, we also showed that Bob can never hope to win the Scoring game. Nevertheless, we identified reflection graphs, a rich family of graphs on which Bob always has a drawing strategy for the Scoring game.

There are several directions to investigate which are linked to this notion of reflection graphs. First, just as reflection graphs define an interesting class of graphs that are AB-draw, another direction could be to find large and interesting classes of graphs that are A-win. Graphs of odd order in which Alice can always construct a single connected red component are A-win, and so, perhaps a class of dense graphs of odd order would be a prime candidate. We provide partial results towards this direction in upcoming Chapter 9.

We also wonder about different types of grids. A valid point for considering such graphs is that grids are natural structures to play on in several types of games, as illustrated by Hex. Indeed, in upcoming Chapter 9, we will discuss about king's grids, for which we will provide bounds on c_g when there are two rows and m columns. Note that the case of grids will be a recurring theme in the following chapters. For the moment, notice that Cartesian and king's grids of even order are AB-draw, which follows directly from Proposition 7.2.3 and Theorem 7.2.4. This leaves us with the case of playing the Scoring game on grids of odd order, which can be the subject of a dedicated study.

Finally, the reductions presented in this chapter open some interesting questions to be explored. For example, since the Maker-Breaker game is PSPACE-complete in split graphs by Corollary 7.3.4, and split graphs have diameter at most 3, there is the question of whether it is hard to compute c_g for graphs of diameter 2. Moreover, in Section 7.3, we showed that the Maker-Breaker game remains PSPACE-complete when restricted to various classes of graphs, but we do not know whether the same holds for the Scoring game in those classes of graphs. These results already point to the fact that trying to establish significant differences between the Scoring and the Maker-Breaker games could be an interesting question by itself.

CHAPTER 8

Playing on simple graphs

In this chapter we focus on determining the outcome of the Scoring and the Maker-Breaker games, when these games are played on specific families of graph. We first consider the Scoring game, proving that the outcome can be computed in constant time when the game is played on paths and cycles, and in linear time when played on co-graphs. We proceed by considering the Maker-Breaker game for which we provide a linear-time algorithm for deciding its outcome when played on a $(q, q-4)$-graph, a familiy of graphs which generalises cographs.

In this chapter we consider both the Scoring and the Maker-Breaker games when played on graphs where deciding the outcome can be done efficiently. In Section 8.1 we show that deciding the outcome of the Scoring game can be done in linear time when playing on a graph G that is a path or a cycle. It should be noted that this result is not trivial, despite the simplicity of these graph families. Moreover, considering the Scoring game on paths and cycles is actually helpful in our endeavor to also determine the parameter c_g of these graphs. Indeed, we do manage to determine $c_g(G)$ when G is either a path or a cycle. Then, in Section 8.2, we show that calculating the outcome of the Scoring game on cographs can be done in linear time. As far as the Maker-Breaker game is concerned, we consider it when played in cographs, and, more generally, in $(q, q-4)$-graphs. For both classes of graphs, we prove in Section 8.3 that c_g can be determined in linear time.

8.1 Paths and cycles

In this section, we deal with the case of playing the Scoring game on n-vertex paths $P_n = (v_1, \ldots, v_n)$ and n-vertex cycles $C_n = (v_1, \ldots, v_n)$. Recall that every path and cycle of even order is a reflection graph by Proposition 7.2.3, and thus, is AB-draw by Theorem 7.2.4. Here, we finish the case of paths and cycles by dealing with the case of paths and cycles of odd order. As a direct corollary, we also compute the value of $c_g(G)$ when G is a path or a cycle.

We begin with two technical lemmata for specific cases in paths, which will be used in the proofs for paths and cycles of odd order. In the following proofs in this section, we often divide the main path P_n into two subpaths Q and Q', and say that Alice "follows" Bob, that is, when Bob plays in Q (in Q', resp.), Alice then plays in the same subpath Q (in Q', resp.). The precise way Alice answers to Bob's moves in Q (in Q', resp.) is described in the proofs and depends on the different cases. Note that, when following this strategy, Alice may be unable to colour a desired vertex (either because Q, resp., Q', has no uncoloured vertex anymore, or because the desired vertex is already coloured red). In such a case, Alice colours any arbitrary uncoloured vertex of the main path. The same applies for when we say that Bob "follows" Alice. Lastly, for any path P_n, let us orient the path from left to right (from its end v_1 to its other end v_n), so that we can make use of the notions of left and right.

The next two lemmata (Lemmata 8.1.1 and 8.1.2) are both stated using first and second player rather than Alice and Bob since they will sometimes be used with Alice as the first player and sometimes with Bob as the first player.

Lemma 8.1.1. *For all $n \geq 1$, for the path P_n, the second player has a strategy that ensures that the largest connected subgraph of the first player is of order at most 2, even if one of the ends of P_n is initially coloured by the first player and it is the first player's turn.*

Proof. Assume, w.l.o.g., that Alice is the first player, Bob is the second player, and v_1 is initially coloured red. Whenever Alice colours a vertex v_j with $2 \leq j \leq n$, Bob colours v_{j-1} if it is uncoloured. If v_{j-1} is already coloured, then Bob colours the closest (in terms of its distance in the path) uncoloured vertex that is to the right of v_j. Towards a contradiction, assume that there exist 3 consecutive red vertices, denoted by x_1, x_2, x_3 from left to right in P_n. By Bob's strategy, concerning the 3 vertices x_1, x_2, x_3, Alice must have coloured x_1 first, then x_2, and then, x_3, as otherwise, Bob would have coloured at least one of them. But when Alice colours x_2, since x_1 is already coloured, then Bob will colour the closest uncoloured vertex to the right of x_2, which must be x_3 since it is uncoloured as it must get coloured by Alice after she colours x_2, and thus, we have a contradiction. □

Lemma 8.1.2. *Let $x \geq 1$ and $n \geq x$. Consider any path P_n with x vertices initially coloured by the second player, and let y be the maximum order of an initial connected component of the second player.*

- *If $y = x$ and, either the component of the second player contains no ends of P_n or $x = 1$, then, if the first player starts, they have a strategy ensuring that the second player cannot create a connected component of order more than $x + 1$;*
- *otherwise, if the first player starts, then they have a strategy ensuring that the second player cannot create a connected component of order more than x.*

Proof. Assume, w.l.o.g., that Alice is the first player and Bob is the second player. We prove the lemma by induction on x. First, let us consider the case $x = 1$. We prove this case by induction on n. If $n = 1$, then the result is obvious, so let us focus on the general case. Without loss of generality, let v_j ($1 \leq j < n$) be the vertex initially coloured blue. Then, Alice first colours v_{j+1}. Let $Q = (v_1, \ldots, v_j)$ and $Q' = (v_{j+2}, \ldots, v_n)$ (it may be that Q' is empty and/or Q is restricted to one vertex). From now on, Alice "follows" Bob, that is, when Bob plays in Q (in Q', resp.),

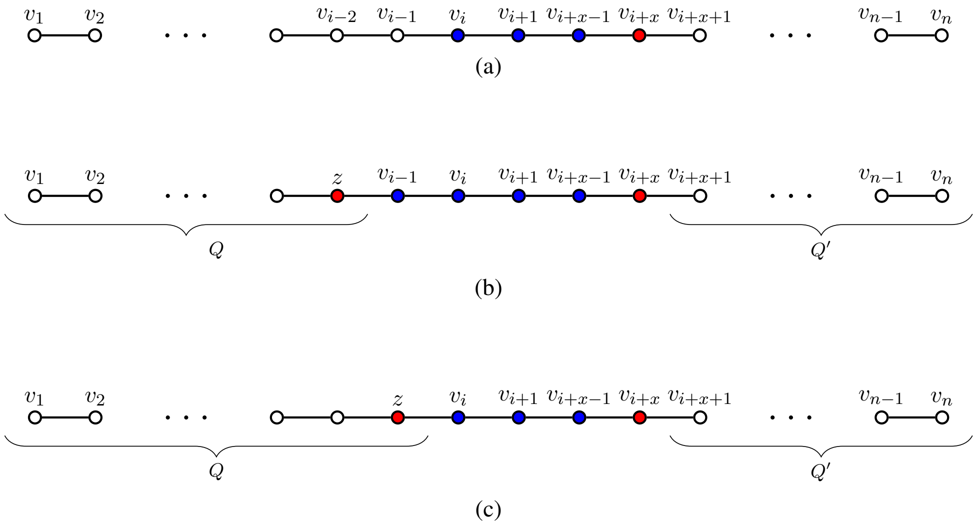


Figure 8.1 – Case in the proof of Lemma 8.1.2 where $y = x = 3$, $B = (v_i, \ldots, v_{i+x-1})$ and contains no ends of P_n, and $i > 2$ (first case when $x > 1$). On her first turn, depicted in (a), Alice colours v_{i+x}. If Bob creates a connected component of order $x + 1$ by colouring v_{i-1} on his next turn, then Alice colours $z = v_{i-2}$, as depicted in (b). Otherwise, Alice colours $z = v_{i-1}$, as shown in (c).

Alice then plays in Q (in Q', resp.), and both games are considered independently (since v_{j+1} is coloured red). Considering Q as a path with one of its ends initially coloured blue, and applying Lemma 8.1.1 to it (with Bob as the first player), Alice has a strategy ensuring that Bob cannot create a connected blue component of order more than 2 in Q. On the other hand, after the first move of Bob in Q', it is a path of order less than n with one vertex initially coloured blue and it is the turn of Alice. Thus, by induction (on n), Alice has a strategy ensuring that Bob cannot create a connected blue component of order more than 2 in Q'. Overall, Alice ensures that the largest connected blue component has order at most $2 = x + 1$. Hence, the claim holds for $x = 1$.

Let $x > 1$ and let us assume by induction that the previous statement holds for all $x' < x$.

— Let us first assume that $y = x > 1$ and the connected blue component B contains no ends of P_n, say $B = (v_i, \ldots, v_{i+x-1})$, $1 < i < n - x + 1$. Alice first colours v_{i+x} (see Figure 8.1(a)). If Bob colours v_{i-1} on his next turn (in which case there is a connected blue component of order $x + 1$), then Alice colours $z = v_{i-2}$ (unless $i = 2$, in which case Alice colours any arbitrary uncoloured vertex). Otherwise, Alice colours $z = v_{i-1}$ (in which case the largest connected blue component is of order x). Let $Q = (v_1, \ldots, z)$ and $Q' = (v_{i+x+1}, \ldots, v_n)$ (it may be that Q and/or Q' are empty, and, in particular, Q is empty if $z \notin \{v_{i-2}, v_{i-1}\}$). See Figures 8.1(b) and 8.1(c) for an illustration of the current configuration of coloured vertices. From now on, Alice "follows" Bob, that is, when Bob plays in Q (in Q', resp.), Alice then plays in Q (in Q', resp.), and both games are considered independently (since z and v_{i+x} are coloured red). After the next move of Bob in Q (Q', resp.), it is a path of order less than n with at most $2 \leq x$ vertices initially coloured blue and it is Alice's turn. Thus, by induction (on n), Alice has a strategy ensuring that Bob cannot create a connected blue component of order more than $x + 1$ in Q (Q', resp.).

Overall, Alice ensures that the largest connected blue component in P_n is of order at most $x+1$. Hence, the claim holds in this case.

— Next, let us assume that $y = x > 1$ and the connected blue component B contains one end of P_n, *i.e.*, $B = (v_1, \dots, v_x)$. Alice first colours v_{x+1}. Then, Bob colours any vertex in the subpath $Q = (v_{x+2}, \dots, v_n)$ (assuming Q is not empty). Therefore, Q initially has one blue vertex and it is Alice's turn. By the base case of the induction ($x = 1$), Alice can ensure that the largest connected blue component in Q is of order at most 2. Overall, the largest connected blue component in P_n is of order at most x. Hence, the claim holds in this case.

— Finally, let us assume that $y < x$. Let $(v_i, \dots, v_{i+y-1})$ be a largest connected blue component such that there is an initial blue vertex v_j with $j > i + y$. Alice first colours v_{i+y}. Let $Q = (v_1, \dots, v_{i+y-1})$ and $Q' = (v_{i+y+1}, \dots, v_n)$ (it may be that Q' is empty). From now on, Alice "follows" Bob, that is, when Bob plays in Q (in Q', resp.), Alice then plays in Q (in Q', resp.), and both games are considered independently (since v_{i+y} is coloured red). After the next move of Bob in Q (Q', resp.), it is a path of order less than n with at most $y+1 \leq x$ vertices initially coloured blue (and if there is a connected blue component with x vertices, it must be in Q and it contains the end v_{i+y-1} of the path Q) and it is Alice's turn. Thus, by induction (on n), Alice has a strategy ensuring that Bob cannot create a connected blue component of order more than x in Q (Q', resp.). Overall, Alice ensures that the largest connected blue component in P_n is of order at most x. Hence, the claim holds in this case, and in general, since this is the last case.

□

We can now deal with the general case of paths of odd order.

Theorem 8.1.3. *For all $n \geq 1$, the path P_n is A-win if and only if $n \in \{1, 3, 5, 7, 9\}$.*

Proof. Note that, by Theorem 7.2.1, we need to prove that P_n is A-win if $n \in \{1, 3, 5, 7, 9\}$, and P_n is AB-draw otherwise. Let $P_n = (v_1, \dots, v_n)$. If n is even, then P_n is AB-draw by Theorem 7.2.4 since P_n is a reflection graph by Proposition 7.2.3. It is easy to see that, if $n \leq 7$ and n is odd, then Alice wins by first colouring the center of P_n. If $n = 9$, a winning strategy for Alice is described in Figure 8.2. Hence, from now on, let us assume that $n \geq 11$ is odd.

In the main strategy that follows, we require that there are at least five vertices to the left or to the right of the first vertex Alice colours, and that is why it does not apply to the paths of odd order less than 11. We will now describe a strategy for Bob which ensures a draw.

Let v_j, with $1 \leq j \leq n$, be the first vertex coloured by Alice. Since $n \geq 11$, there are at least five vertices to the left or right of v_j, say, w.l.o.g., to the left of v_j, *i.e.*, $6 \leq j \leq n$. Bob colours v_{j-1}. Let $Q = (v_1, \dots, v_{j-1})$ and $Q' = (v_j, \dots, v_n)$. From now on, Bob "follows" Alice, that is, when Alice plays in Q (in Q', resp.), Bob then plays in Q (in Q', resp.), and both games are considered independently (since v_{j-1} is coloured blue and v_j is coloured red). Considering Q' as a path with one of its ends initially coloured red, and applying Lemma 8.1.1 to it (with Alice as the first player), Bob has a strategy ensuring that Alice cannot create a connected red component of order more than 2 in Q'. Let v_ℓ be the first vertex that Alice colours in Q. We distinguish two cases:

1. $\ell \neq j-2$. Then, Bob colours v_{j-2}. During the next rounds, whenever Alice plays in Q, while it is possible, Bob colours a neighbour of the connected blue component containing v_{j-1} and v_{j-2}. When it is not possible anymore, either the connected blue component is of

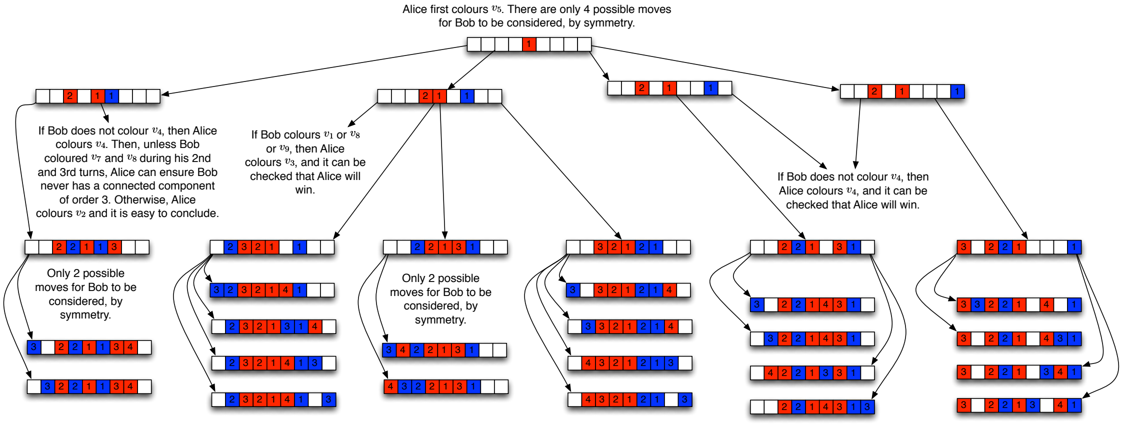


Figure 8.2 – Winning strategy for Alice in P_9. The squares represent the vertices v_1 to v_9 from left to right. A number i in a red (blue, resp.) square indicates this vertex is the i^{th} vertex coloured by Alice (Bob, resp.). Each arrow corresponds to a move of Bob and then one of Alice. The last moves in each case are omitted as it is easy to check the last possibilities.

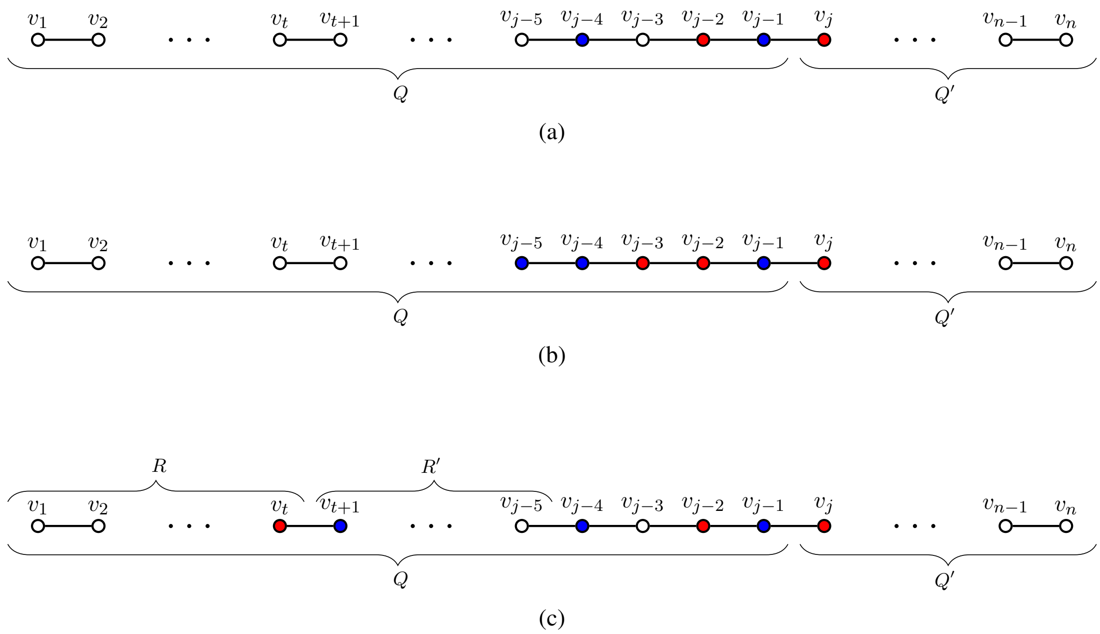


Figure 8.3 – Second case in the proof of Theorem 8.1.3 where $\ell = j-2$. Bob begins by colouring v_{j-4}, as shown in (a). Figure (b) illustrates the case in which Alice then colours v_{j-3}. Figure (c) illustrates the case in which Alice then colours v_t for $1 \leq t \leq j-6$ (in the illustration, $t < j-6$).

order $\lceil (j-1)/2 \rceil \geq 3$ (in which case the largest connected red component in Q is of order $\lfloor (j-1)/2 \rfloor$ and so, Alice does not win) or it is of order x with $2 \leq x < (j-1)/2$ and it is Bob's turn. In the latter case, the connected blue component in Q consists of the vertices $v_{j-x}, \ldots, v_{j-1}$, and v_{j-x-1} is red since Bob cannot colour a neighbour of the connected blue component. Let $R = (v_1, \ldots, v_{j-x-1})$ and note that there are exactly x red vertices in R including v_{j-x-1} (one of its ends). Then, applying Lemma 8.1.2 to R (with Bob as the first player), Bob has a strategy ensuring that Alice cannot create a connected red component of order more than x in R. As usual, whenever Bob cannot follow his strategy, he simply colours any arbitrary vertex. Hence, the game in P_n ends in a draw in this case.

2. $\ell = j - 2$. Then, Bob colours v_{j-4} (illustrated in Figure 8.3(a)). Now, if Alice colours v_{j-3}, then Bob colours v_{j-5} (as shown in Figure 8.3(b)), and *vice versa*, and this guarantees that there is a connected blue component of order at least 2. Otherwise, if Alice colours a vertex v_t with $1 \leq t \leq j-6$, then Bob colours v_{t+1}, unless v_{t+1} is already coloured, in which case, Bob colours v_{t-1}. In the latter case, Bob can ensure a draw since he can ensure that Alice cannot create a connected red component of order more than 2 in $R^* = (v_1, \dots, v_{t-1})$ by Lemma 8.1.1 (with Alice as the first player). So, assume we are in the former case. Let $R = (v_1, \dots, v_t)$ and $R' = (v_{t+1}, \dots, v_{j-5})$ (see Figure 8.3(c)). From now on, Bob "follows" Alice (unless Alice colours v_{j-5}, in which case, Bob colours v_{j-3}), that is, when Alice plays in R (in R', resp.), Bob then plays in R (in R', resp.), and both games are considered independently (since v_t is coloured red and v_{t+1} is coloured blue). Considering R as a path with one of its ends initially coloured red, and applying Lemma 8.1.1 to it (with Alice as the first player), Bob has a strategy ensuring that Alice cannot create a connected red component of order more than 2 in R. Bob plays in R' assuming that v_{j-5} is already coloured red, and applying Lemma 8.1.1 to it (with Alice as the first player), Bob has a strategy ensuring that Alice cannot create a connected red component of order more than 2 in R'. As usual, whenever Bob cannot follow his strategy, he simply colours any arbitrary vertex. It is easy to see that, in this case, the largest connected blue (red, resp.) subgraph is of order at least 2 (at most 2, resp.).

This concludes the proof for $n \geq 11$. □

Now, we address the Scoring game in cycles. We again start with a technical lemma for a specific case in paths, which we will use in the proof for cycles.

Lemma 8.1.4. *Let $x \geq 3$, $n \geq x+1$, and let $n-x$ be odd. Consider any path P_n with x vertices, including both ends, initially coloured blue. If Alice starts, then she has a strategy ensuring that Bob cannot create a connected blue component of order more than $x-1$ in P_n.*

Proof. The first case, $x = 3$, is proven by induction on n. If $n = 4$, the result obviously holds, so assume that $n > 4$ and that the induction holds for all $n' < n$.

— First, assume that the initial blue vertices are v_1, v_2, and v_n. Then, Alice colours v_3. Then, Bob colours any uncoloured vertex in $Q = (v_4, \dots, v_n)$. Now, Q has two blue vertices (and if there is a connected blue component of order 2 in Q, it contains the end v_n of Q). By Lemma 8.1.2 (with Alice as the first player), Alice can ensure that Bob cannot create a connected blue component with more than two vertices in Q. Overall, Bob cannot create a connected blue component of order at least 3 in P_n.

— Next, let v_1, v_j, v_n (with $2 < j < n-1$) be the initial blue vertices. W.l.o.g. (up to reversing the path), assume that j is even (note that n is even since $n - x = n - 3$ is odd). Then, Alice colours v_{j+1}. Let $Q = (v_1, \dots, v_j)$ and $Q' = (v_{j+2}, \dots, v_n)$ (it may be that Q' is just the vertex v_n). From now on, Alice "follows" Bob, that is, when Bob plays in Q (in Q', resp.), Alice then plays in Q (in Q', resp.), and both games are considered independently (since v_{j+1} is coloured red). For the game in Q', applying Lemma 8.1.1 (with Bob as the first player), Alice can ensure the largest connected blue component is of order at most 2 in Q'. For the game in Q, by induction on $n' = |Q| < n$ (note that, because $n' = j$ is even, after the first turn of Bob in Q, the hypotheses hold for $x = 3$ in Q), Alice can ensure the largest connected blue component is of order at most 2 in Q. Overall, Bob cannot create a connected blue component of order at least 3 in P_n.

Now, let us assume that $x > 3$.

First, if there is a connected blue component of order $x - 1$ containing v_1, then Alice colours v_x, and then she can ensure, by Lemma 8.1.1 (with Bob as the first player), that Bob cannot create a connected blue component with more than two vertices in $(v_{x+1}, \ldots, v_n)$.

Next, assume that there exists a connected blue component $(v_j, \ldots, v_{j+x-3})$ of order $x - 2$ not containing any end of P_n. By symmetry (up to reversing the path), let $j \leq n - j - x + 4$. Alice first colours v_{j-1}. Let $Q = (v_1, \ldots, v_{j-2})$ and $Q' = (v_j, \ldots, v_n)$. From now on, Alice "follows" Bob, that is, when Bob plays in Q (in Q', resp.), Alice then plays in Q (in Q', resp.), and both games are considered independently (since v_{j-1} is coloured red). Since $j \leq n - j - x + 4$, we get that $2j - 4 \leq n - x$. Since $n - x$ is odd, it implies that $2j - 4 \leq n - x - 1$, and so, we get $j \leq n - j - x + 3$. Finally, since $j \geq 3$ (as the connected blue component $(v_j, \ldots, v_{j+x-3})$ does not contain v_1 which is also blue), it follows that $n - j + 1 \geq x + 1$. Therefore, $Q' = (v_j, \ldots, v_n)$ is of order at least $x + 1$. When Bob plays in Q, Alice can ensure, by Lemma 8.1.1 (with Bob as the first player), that Bob cannot create a connected blue component with more than two vertices in Q. When Bob first plays in Q', then Q' becomes a path of order at least $x + 1$ with x initial blue vertices, and its largest connected blue component contains its end v_j and is of order at most $x - 1$. By Lemma 8.1.2 (with Alice as the first player), Alice can ensure that Bob does not create a connected blue component of order more than $x - 1$ in Q'.

Otherwise, there must be an uncoloured vertex v_j such that at most $x - 2$ blue vertices are on the left (on the right, resp.) of v_j. Then, Alice first colours v_j. Let $Q = (v_1, \ldots, v_{j-1})$ and $Q' = (v_{j+1}, \ldots, v_n)$. From now on, Alice "follows" Bob, that is, when Bob plays in Q (in Q', resp.), Alice then plays in Q (in Q', resp.), and both games are considered independently (since v_j is coloured red). By Lemma 8.1.2 (with Alice as the first player), Alice can ensure, both in Q and Q', that Bob does not create a connected blue component with at least x vertices (note that after the first turn of Bob in Q (Q', resp) it contains at most $x - 1$ blue vertices including at least one of its ends). □

Theorem 8.1.5. *For all $n \geq 3$, the cycle C_n is A-win if and only if n is odd.*

Proof. If n is even, then C_n is a reflection graph by Proposition 7.2.3, and so, is AB-draw by Theorem 7.2.4. Let us now assume that n is odd. We describe a winning strategy for Alice. If $n \leq 5$, the result is obvious, so let us assume that $n > 5$.

First, let us assume (independently of how this configuration eventually appears) that after $x \geq 3$ rounds, the vertices $v_1, \ldots, v_x$ have been coloured red, the vertices v_n and v_{x+1} are coloured blue, and any $x - 2$ other vertices in $\{v_{x+2}, \ldots, v_{n-1}\}$ are coloured blue (see Figure 8.4 for an illustration). Note that it is now Alice's turn. By Lemma 8.1.4, Alice may ensure that Bob cannot create a connected blue component of order at least x in the subgraph induced by $(v_{x+1}, \ldots, v_n)$. Therefore, in that situation, Alice wins.

Now, let Alice first colour the vertex v_1. If Bob does not colour a neighbour of v_1 (say Bob colours v_j with $3 < j < n$, since $n > 5$), then, on her second turn, Alice colours v_2. During the next rounds, while it is possible, Alice colours a neighbour of the connected red component. When

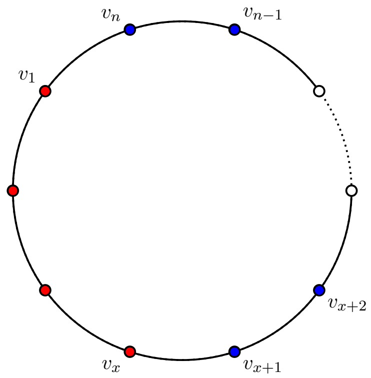


Figure 8.4 – First case in the proof of Theorem 8.1.5 where, after $x = 4$ rounds, the vertices $v_1, \dots, v_x$ are red, the vertices v_n and v_{x+1} are blue, and $x-2$ other vertices in $\{v_{x+2}, \dots, v_{n-1}\}$ are blue (in this illustration, these vertices are v_{n-1} and v_{x+2}).

it is not possible anymore, either the connected red component is of order $\lceil n/2 \rceil$ or it is of order at least 3 and we are in the situation of the above paragraph. In both cases, Alice wins.

Therefore, after Alice colours her first vertex (call it v_2), Bob must colour some neighbour of it (say v_1). By induction on the number $t \geq 1$ of rounds, let us assume that the game reaches, after t rounds, a configuration where, for every $1 \leq i \leq t$, the vertices v_{2i-1} are coloured blue and the vertices v_{2i} are coloured red. If $t = \lfloor n/2 \rfloor$, then Alice finally colours v_n (recall that n is odd) and wins. Otherwise, let Alice colour v_{2t+2}.

— If Bob then colours v_{2t+1}, then we are back to the previous situation for $t' = t+1$. Then, eventually, Alice wins by induction on $n - 2t$.

— If Bob does not colour v_{2t+1}, then Alice colours v_{2t+1} and then continues to grow the connected red component containing v_{2t+1} while possible. When it is not possible anymore, note that contracting the vertices v_2 to v_{2t} to a single red vertex, we are back to the situation of the first paragraph of this proof (with a connected red component of order at least 3) and, therefore, Alice wins.

□

Allow us to close this section by exhibiting how the arguments used in the proofs above, and in particular Lemma 8.1.1, can also be useful for providing results when considering the Maker-Breaker game.

Proposition 8.1.6. *Let G be either a path or a cycle of order at least 3. Then $c_g(G) = 2$.*

Sketch of the proof. The case where G is a path of order at least 3 is a direct corollary of Lemma 8.1.1. We will now show that the statement holds when $G = C_n$, for $n \geq 3$.

We first show that $c_g(C_n) \geq 2$. This is quite trivial as it suffices for Alice to colour any vertex v of C_n on her first turn, and any uncoloured vertex $u \in N(v)$ on her second turn. Note that u always exists as $N(v) = 2$ and Bob only colours one vertex in the meantime.

To show that $c_g(C_n) \leq 2$, Bob's strategy is to colour a vertex adjacent to the red vertex in the first round, and now, the game is equivalent to one on a path P_{n-1}, with one of its ends initially coloured red. The result follows by Lemma 8.1.1. ◇

In the following proposition we use the strategies we described for the Scoring game when played on a path, in order to provide an upper bound on the value of $c_g(G)$ when G is a Cartesian grid, *i.e.*, the Cartesian product $P_n \square P_m$ of the paths P_n and P_m.

Proposition 8.1.7. *Let* $n \leq m$. *Then,* $c_g(P_n \square P_m) \leq 2n$.

Sketch of the proof. Let us consider an $n \times m$ grid $P_n \square P_m$ with n rows and m columns (with left and right being defined naturally). Let us consider the following strategy for Bob. When Alice colours a vertex v, if the right neighbour u of v exists and is uncoloured, then Bob colours u, otherwise, Bob colours the left neighbour of v if it exists and is uncoloured, and otherwise, Bob colours any arbitrary uncoloured vertex.

The above strategy for Bob is well-defined and ensures that no three consecutive vertices in a row are ever red (see the case of paths in Section 8.1 for more details). This ensures that, for any strategy of Alice, any connected red component has at most 2 vertices in each row, hence, proving the lemma. Indeed, consider a largest connected red component S at the end of the game. Towards a contradiction, assume that there exists a row whose intersection with S contains strictly more than 2 vertices. Then, the restriction of S to this row induces at least two connected red components X and Y (since there cannot be three consecutive red vertices in a same row). Let P be any red path from X to Y (that exists since S is connected). It can be shown that P must contain 3 consecutive red vertices in a same row, a contradiction. ◇

8.2 The Scoring game on cographs

We continue focusing on the Scoring game. In the previous case of paths and cycles, presented in Section 8.1, we have seen examples where playing optimally depends more on positional play with respect to the previously coloured vertices and the graph's properties, since the sparse structure of the graph makes it very easy for the players to stop the expansion of the opponent's largest connected component. As a consequence, in such cases it is likely that the players must, at some point, stop growing their largest connected component, and start growing a new one. Obviously, such a strategy is likely to be far less viable in graph classes that are denser. In such denser graphs, the game actually tends to turn into a rather different one, where the players grow a single connected component each, that they have to keep "alive" for as long as possible. We illustrate these thoughts with the case of cographs, which leads us to introduce a few more notations to describe a linear-time algorithm deciding the outcome of the Scoring game in such instances.

Definition 8.2.1. *A graph G is a* cograph *if it is P_4-free, i.e., if it does not contain any path with four vertices as an induced subgraph. The class of cographs can also be defined recursively as follows. The one-vertex graph K_1 is a cograph. Let G_1 and G_2 be two cographs. Then, the disjoint union $G_1 + G_2$ is a cograph. Moreover, the join $G_1 \oplus G_2$, obtained from $G_1 + G_2$ by adding all the possible edges between the vertices of G_1 and the vertices of G_2, is also a cograph.*

Note that a decomposition of a cograph (*i.e.*, a building sequence of unions and joins performed from single vertices) can be computed in linear time [56].

To simplify the notation in the theorem and its proof to follow, let us introduce the following graph family.

Definition 8.2.2. *A graph G is A-*perfect *if there exists a strategy for Alice that ensures a connected red component of order $\left\lceil \frac{|V(G)|}{2} \right\rceil$, regardless of Bob's strategy. That is, if a graph G is A-perfect, then Alice has a strategy to ensure a single connected red component when playing either the Scoring or the Maker-Breaker game on G.*

Theorem 8.2.3. *Let G be a cograph. There exists a linear-time algorithm that decides whether G is A-win or AB-draw, and whether G is A-perfect or not.*

Proof. The proof is by induction on $n = |V(G)|$. More precisely, we describe a recursive algorithm. If $n = 1$, then G is clearly A-win and G is A-perfect.

Let us now assume that $n > 1$. There are two cases to be considered. Either $G = G_1 \oplus G_2$ for some cographs G_1 and G_2, or $G = G_1 + \ldots + G_m$, where, for every $1 \leq i \leq m$ ($m \geq 2$), G_i is either a single vertex or is a cograph obtained from the join of two other cographs. For every $1 \leq i \leq m$, let us assume by induction that it can be computed in time linear in $|V(G_i)|$, whether G_i is A-win or AB-draw and whether G_i is A-perfect or not. Let us show, now, how to decide if G is A-win or AB-draw, and whether G is A-perfect or not, in constant time.

1. Let us first assume that $\boldsymbol{G = G_1 \oplus G_2}$. There are three cases to be distinguished.

   (a) If n is odd (so we may assume that $|V(G_2)| \geq 2$), then G is A-win and G is A-perfect. We describe a winning strategy for Alice. Alice first colours a vertex in G_1. In the second round, Alice colours a vertex in G_2 (it is possible since $|V(G_2)| \geq 2$). Then, Alice colours any uncoloured vertex in each of the remaining rounds. Regardless of Bob's strategy, Alice ends with all the $\lceil \frac{n}{2} \rceil$ red vertices belonging to the same connected component. Since n is odd, G is A-win, and G is A-perfect.

   (b) If $|V(G_1)|, |V(G_2)| \geq 2$ and n is even, then G is AB-draw and G is A-perfect. We describe a drawing strategy for Bob. W.l.o.g., Alice first colours a vertex in G_1. Then, Bob first colours a vertex in G_1 (it is possible since $|V(G_1)| \geq 2$). In the second round, Bob colours a vertex in G_2 (it is possible since $|V(G_2)| \geq 2$). Then, Bob colours any uncoloured vertex in each of the remaining rounds. Regardless of Alice's strategy, Bob ends with all the $n/2$ blue vertices belonging to the same connected component. Since n is even, Alice cannot have a larger connected red component. Hence, G is AB-draw and G is A-perfect.

   (c) Finally, let us assume that $|V(G_1)| = 1$ (let u be the single vertex of G_1) and n is even (so $|V(G_2)|$ is odd). There are two cases to be considered.

      i. If G_2 is not A-perfect, then G is A-win and G is A-perfect. We describe a winning strategy for Alice. Alice first colours u. Then, she plays in G_2 as the second player, and thus, she can ensure that any connected blue component is of order less than $\left\lceil \frac{|V(G_2)|}{2} \right\rceil = \lceil \frac{n-1}{2} \rceil$ in G_2 since G_2 is not A-perfect. Since u is a universal vertex, regardless of Bob's strategy, Alice ensures a connected red component of order $n/2$, and so G is A-win and G is A-perfect.

      ii. If G_2 is A-perfect, then G is AB-draw and G is A-perfect. We describe a drawing strategy for Bob. If Alice first colours a vertex of G_2, then Bob colours u, and then Bob colours any uncoloured vertex of G_2 in each of the subsequent rounds. Then, Bob ensures a connected blue component of order $n/2$, and so G is AB-draw and G is A-perfect.

Otherwise, if Alice starts by colouring u, then Bob can play as the first player in G_2 and, in doing so, ensure a connected blue component of order $\lceil \frac{n-1}{2} \rceil = n/2$ in G_2. Then, again G is AB-draw and G is A-perfect.

2. Now, let us assume that $\boldsymbol{G = G_1 + \ldots + G_m}$ where, for every $1 \leq i \leq m$ ($m \geq 2$), G_i is either a single vertex or is a cograph obtained from the join of two other cographs. For all $1 \leq i \leq m$, if G_i is a cograph obtained from the join of two other cographs, then let those two cographs be G'_i and G''_i, and let $|V(G'_i)| \geq |V(G''_i)|$. Also, let $n_i = |V(G_i)|$ for every $1 \leq i \leq m$, and let us assume that $n_1 \geq \ldots \geq n_m$.

   To simplify the case analysis to follow, we will show that we can make several assumptions. First note that, if $n_1 = 1$, then G is AB-draw (since $n_2 = 1$ as $m \geq 2$) and G is A-perfect if and only if $G = G_1 + G_2$. Hence, we may assume that $n_1 > 1$. Second, if $n_2 = 1$, then the result of the game in G is the same as the result of the game in G_1, and this result is known since G_1 is a join (recall Case 1 of the proof). Moreover, in this case, G is A-perfect if and only if n_1 is odd and $G = G_1 + G_2$. Hence, we may also assume that $n_2 > 1$. Lastly, in what follows, for any of the winning strategies described for Alice, whenever Bob colours a vertex in G_j for $3 \leq j \leq m$, Alice also colours a vertex in G_j on her next turn. The same holds for any of the drawing strategies for Bob (with Bob and Alice reversed), except for Case 2(e)ii, in which case the same only holds for $4 \leq j \leq m$. This guarantees that a player never has a connected component of order more than $\lceil \frac{n_j}{2} \rceil$ in G_j for $3 \leq j \leq m$ (except for Case 2(e)ii in which case the same only holds for $4 \leq j \leq m$). Let us remark that Alice will always have a connected red component of order at least $\lceil \frac{n_1}{2} \rceil$ in all of the winning strategies described for Alice below, and Bob will always have a connected blue component of order at least $\lceil \frac{n_1}{2} \rceil$ in all of the drawing strategies described for Bob below. Hence, for all of the cases except Case 2(e)ii, we can assume that $G = G_1 + G_2$, and for Case 2(e)ii, we can assume that $G = G_1 + G_2 + G_3$. In what follows, if a player cannot follow their strategy in a round, unless otherwise stated, they simply colour any arbitrary vertex in that round and then resume their strategy for the subsequent rounds.

   There are five cases to be considered, and recall that we assume that $n_1 > 1$ and $n_2 > 1$ as stated above, which implies that G''_1 and G''_2 exist. Note also that in Case 2(e)iii below, the statement involves n_3, which is not defined if $m = 2$; in such cases, we consider that $n_3 = 0$, *i.e.*, regard G_3 as an empty graph. Moreover, since Bob always has a strategy where, for each $1 \leq i \leq m$, he colours at least $\lfloor \frac{n_i}{2} \rfloor$ vertices of G_i blue, and since $n_2 > 1$, then G is not A-perfect in all five of the following cases. Thus, all that remains to show is the outcome of the game on G for each of the cases.

   (a) If $n_1 = n_2$, then G is AB-draw.

   We describe a drawing strategy for Bob. Assume, w.l.o.g., that Alice first colours a vertex in G_1. Bob then colours a vertex in G''_2. Then, whenever Alice colours a vertex in G_1 (G_2, resp.), Bob also colours a vertex in G_1 (G_2, resp.). In particular, if Bob is to colour a vertex in G_2, then he colours one in G'_2 first if possible, and if not, then he colours a vertex in G''_2, and lastly, if that is not possible, he colours a vertex in G_1. Similarly, if Bob is to colour a vertex in G_1 by this strategy, but cannot since all of the vertices of G_1 are coloured, then he colours one in G'_2 first if possible, and if not, then he colours a vertex in G''_2.

If n_1 is odd, then by this strategy, Bob ensures a connected blue component of order $\frac{n_2-1}{2}+1=\frac{n_1-1}{2}+1$ in G_2 and that the largest connected red component in G is of order at most $\frac{n_1-1}{2}+1$.

If n_1 is even, then by this strategy, if Alice colours the last vertex in G_1, then Bob ensures a connected blue component of order $\lceil\frac{n_2-1}{2}\rceil+1=\lceil\frac{n_1-1}{2}\rceil+1$ in G_2 and that the largest connected red component in G is of order at most $\lceil\frac{n_1-1}{2}\rceil+1$. If, on the other hand, Alice did not colour the last vertex in G_1, and so, she coloured the last vertex in G_2, then Bob ensures a connected blue component of order $\lceil\frac{n_2-2}{2}\rceil+1=\frac{n_1}{2}$ in G_2 and that the largest connected red component in G is of order at most $\frac{n_1}{2}$. Hence, G is AB-draw.

(b) If $n_1>n_2$ and n_1 is odd, then G is A-win.

We describe a winning strategy for Alice. Alice first colours a vertex in G_1. Then, whenever Bob colours a vertex in G_1 (G_2, resp.), Alice colours a vertex in G_1 (G_2, resp.). By Case 1(a), Alice has a winning strategy in G_1 ensuring a connected red component of order at least $\lceil\frac{n_1}{2}\rceil$. By Case 1, Alice ensures that any connected blue component in G_2 is of order at most $\lceil\frac{n_2}{2}\rceil<\lceil\frac{n_1}{2}\rceil$. Hence, G is A-win.

(c) If $n_1>n_2$, n_1 is even, and $|V(G_1'')|\geq 2$, then G is AB-draw.

We describe a drawing strategy for Bob. Whenever Alice colours a vertex in G_1 (G_2, resp.), Bob also colours a vertex in G_1 (G_2, resp.). By Case 1(b), Bob has a drawing strategy in G_1 ensuring a connected blue component of order at least $\frac{n_1}{2}$. By Case 1, Bob ensures that any connected red component in G_2 is of order at most $\lceil\frac{n_2}{2}\rceil\leq\frac{n_1}{2}$. Hence, G is AB-draw.

(d) If $n_1>n_2$, n_1 is even, $|V(G_1'')|=1$, and G_1' is A-perfect, then G is AB-draw.

We describe a drawing strategy for Bob. Whenever Alice colours a vertex in G_1 (G_2, resp.), Bob also colours a vertex in G_1 (G_2, resp.). By Case 1(c)ii, Bob has a drawing strategy in G_1 ensuring a connected blue component of order at least $\frac{n_1}{2}$. By Case 1, Bob ensures that any connected red component in G_2 is of order at most $\lceil\frac{n_2}{2}\rceil\leq\frac{n_1}{2}$. Hence, G is AB-draw.

(e) If $n_1>n_2$, n_1 is even, $|V(G_1'')|=1$, and G_1' is not A-perfect, then there are three cases to be considered.

i. If $n_1>n_2+1$, then G is A-win.

We describe a winning strategy for Alice. Alice first colours a vertex in G_1. Then, whenever Bob colours a vertex in G_1 (G_2, resp.), Alice colours a vertex in G_1 (G_2, resp.). By Case 1(c)i, Alice has a winning strategy in G_1 ensuring a connected red component of order at least $\frac{n_1}{2}$, and that any connected blue component in G_1 is of order less than $\frac{n_1}{2}$. By Case 1, Alice ensures that any connected blue component in G_2 is of order at most $\lceil\frac{n_2}{2}\rceil<\frac{n_1}{2}$. Hence, G is A-win.

ii. If $n_1=n_2+1=n_3+1$, then G is AB-draw.

We describe a drawing strategy for Bob. Whenever Alice colours a vertex in G_1, Bob also colours a vertex in G_1. By Case 1, this ensures that $\frac{n_1}{2}$ of the vertices in G_1 are red and $\frac{n_1}{2}$ of them are blue. The first time that Alice colours a vertex $v\in V(G_2)\cup V(G_3)$, assume, w.l.o.g., that $v\in V(G_2)$. Bob then colours a vertex in G_3''. Then, whenever Alice colours a vertex in G_2 (G_3, resp.), Bob also colours a vertex in G_2 (G_3, resp.). In particular, if Bob is to colour a vertex in G_3, then he colours one in G_3' first if possible, and if not, then he colours a vertex in G_3'', and lastly, if that is not possible, he colours a vertex in G_2. As in Case 2(a), by this strategy, Bob ensures a connected blue component of order $\lceil \frac{n_3}{2} \rceil = \frac{n_1}{2}$ in G_3 and that any connected red component in G_2 is of order at most $\lceil \frac{n_2}{2} \rceil = \frac{n_1}{2}$. Hence, G is AB-draw.

iii. If $n_1 = n_2 + 1$ and $n_2 > n_3$, then G is A-win.
We describe a winning strategy for Alice. Alice first colours the vertex in G_1''. Then, Alice colours vertices in G_1 as long as she can. By Case 1(c)i, she ensures that any connected blue component in G_1 is of order less than $\frac{n_1}{2}$. If it is Alice's turn, there is a connected red component of order $n_1 - k$ in G_1 for some $0 \leq k \leq \frac{n_1}{2}$, and it is the first round in which she can no longer colour vertices in G_1, then Bob coloured k vertices in G_1 and $n_1 - 2k$ vertices in G_2. Then, any connected blue component in G_2 is of order at most $\lceil \frac{n_2 - n_1 + 2k - 1}{2} \rceil + n_1 - 2k = n_1 - k - 1 < n_1 - k$. Hence, G is A-win.

The statement of the theorem then follows since a decomposition of a cograph can be computed in linear time. □

In the following section we provide a polynomial-time algorithm that computes the result of the Maker-Breaker game when played on $(q, q-4)$-graphs, which generalise cographs.

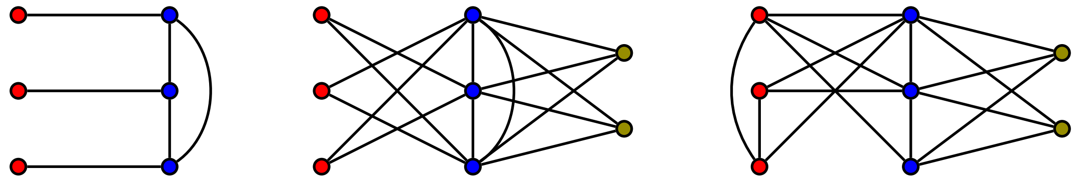
8.3 The Maker-Breaker game on $(q, q-4)$-graphs

In this section, we consider the Maker-Breaker game played in $(q, q-4)$-graphs [12], a graph family that generalises cographs. For a fixed q, these are graphs for which no set of at most q vertices induces more than $q - 4$ P_4 's. Note that cographs are exactly the $(q, q-4)$-graphs for $q = 4$. The study of $(q, q-4)$-graphs is only made possible here thanks to the next lemma which, although relatively straightforward, is rather useful when considering this game. Intuitively, the next lemma shows that when playing the Maker-Breaker game on a disconnected graph G, Alice should focus on the connected component which is the most favourable for her. Note that this is not necessarily the case when playing the Scoring game on G, as can be justified by the proof of Theorem 8.2.3.

Lemma 8.3.1. *If G is a graph with connected components $G_1, \ldots, G_k$, then*

$$c_g(G) = \max \{c_g(G_1), \ldots, c_g(G_k)\}.$$

Proof. To show that $c_g(G) \leq \max \{c_g(G_1), \ldots, c_g(G_k)\}$, it suffices to show that for every subgraph H of G, $c_g(H) \leq c_g(G)$. We give a strategy for Alice ensuring her a score of at least $c_g(H)$ in G. Alice first plays in H according to an optimal strategy $\mathcal{S}$ in H. Then, whenever Bob plays in H, Alice responds in H according to $\mathcal{S}$, and if this is not possible (the vertex to be coloured by $\mathcal{S}$ is already coloured or there are no uncoloured vertices in H) or Bob plays in G, then Alice colours any arbitrary uncoloured vertex in G. In particular, whenever Alice is forced to colour an arbitrary vertex in H, she ignores the fact that vertex is coloured when considering her strategy $\mathcal{S}$ in H in the future. The result follows since Alice will colour at least all the vertices in H that she would colour by $\mathcal{S}$, ensuring her a score of at least $c_g(H)$ in G since $\mathcal{S}$ is optimal in H.



(a) A matched spider (b) An antimatched spider (c) A q-pseudo-spider, $0 \leq q \leq 6$

Figure 8.5 – Examples of graphs described in Definition 8.3.4. The colours in the vertices are used to denote to which set each vertex belongs. The colour red is used for the vertices of S, the colour blue for the vertices of K and olive for the vertices of R.

We will now show that $c_g(G) \geq \max\{c_g(G_1), \ldots, c_g(G_k)\}$. Since the k connected components are pairwise disconnected, Bob can just respond in the same connected component that Alice just played in during each turn. If this is not possible, he colours any arbitrary uncoloured vertex in G, which can only be beneficial to him. □

The main consequence of Lemma 8.3.1, is that when studying c_g for a given class of graphs, we only need to focus on its connected members. Thus, dealing with cographs in the context of the Maker-Breaker game is quite easier than doing so in the context of the Scoring game, as it suffices to focus only on connected cographs, hence, the join of two cographs.

Lemma 8.3.2. *For every two graphs G and H, the join $G \oplus H$ is A-perfect.*

Proof. Consider the strategy for Alice where she aims at having coloured a vertex $u \in V(G)$ and a vertex $v \in V(H)$ by the end of the second round. Note that this is always possible to achieve, unless $|V(G)| = |V(H)| = 1$, in which case the statement is clearly true. In the other cases, $\{u, v\}$ is a connected dominating set of $G \oplus H$, and thus, the result follows by Lemma 7.1.1. □

Theorem 8.3.3. *If G is a cograph, then determining $c_g(G)$ can be done in linear time.*

Proof. Since every connected cograph is the join of two cographs, then, by Lemmata 8.3.1 and 8.3.2, we get that $c_g(G) = \left\lceil \frac{|V(G_i)|}{2} \right\rceil$, where G_i is the connected component of G of the largest order. □

Before we deal with $(q, q-4)$-graphs, let us state the formal definitions of these graphs.

Definition 8.3.4. *Let $G = (S, K, R, E)$ be a graph with $V(G) = S \cup K \cup R$ and $E(G) = E$. Consider the following properties:*

1. *$S \cup K \cup R$ is a partition of $V(G)$ and R can be the empty set.*
2. *$G[K \cup R] = K \oplus R$ (i.e., for all $u, v \in V(G)$ such that $u \in K$ and $v \in R$, we have that $uv \in E$), and K separates S from R (i.e., for all $u \in S$ and $v \in R$, we have that $uv \notin E$).*
3. *S is an independent set, K is a clique, $|S| = |K| \geq 2$, and there exists a bijection $f : S \to K$ such that, either, for every vertex $s \in S$, $N(s) \cap K = K \setminus \{f(s)\}$, or, for every vertex $s \in S$, $N(s) \cap K = \{f(s)\}$. In the former case, we say that f is an antimatching, with the vertices s and $f(s)$ being antimatched, and in the latter case, we say that f is a matching, with the vertices s and $f(s)$ being matched.*

If $G = (S, K, R, E)$ verifies all the properties above, it is called a spider. In that case, if f is a matching (antimatching, resp.), we say that G is a matched spider (antimatched spider, resp.). Also, if G only verifies Properties 1. and 2. above, it is called a pseudo-spider. In this case, for any fixed $q \geq 0$ such that $|V(S \cup K)| \leq q$, we say that G is a q-pseudo-spider. In Figure 8.5 we illustrate examples for such graphs.

For a fixed $q \geq 0$, a graph G is a $(q, q-4)$-*graph* if every subset $S \subseteq V(G)$ of at most q vertices of G induces at most $q-4$ paths on 4 vertices. Note that a cograph is a $(q, q-4)$-graph when $q = 4$. Equivalently:

Theorem 8.3.5 ([13])**.** *A graph G is a $(q, q-4)$-graph if one of the following is satisfied:*

1. *G is the graph K_1.*
2. *$G = G_1 + G_2$, where G_1 and G_2 are $(q, q-4)$-graphs.*
3. *$G = G_1 \oplus G_2$, where G_1 and G_2 are $(q, q-4)$-graphs.*
4. *G is the spider (S, K, R, E), where $G[R]$ (if R is not empty) is a $(q, q-4)$-graph. Note that, by the definition of a spider, $G[S \cup K]$ induces a $(q, q-4)$-graph.*
5. *G is the q-pseudo-spider (S, K, R, E), where $G[R]$ (if R is not empty) is a $(q, q-4)$-graph.*

The above theorem gives us a recursive definition of $(q, q-4)$-graphs. In particular, for every $(q, q-4)$-graph G, there exists a decomposition-tree (not necessarily unique) representing G. The internal nodes of such a decomposition correspond to subgraphs of G that are $(q, q-4)$-graphs, and its leaves either correspond to a single vertex or to a subgraph with at most q vertices. The root corresponds to G, and every internal node has exactly two children (describing the four cases 2 to 5 above). Such a decomposition-tree can be computed in linear time [14]. We are now ready to prove the main result in this section:

Theorem 8.3.6. *Let $q \geq 0$. For a $(q, q-4)$-graph G, determining $c_g(G)$ and an optimal strategy for Alice can be done in linear time.*

Proof. Let us first compute (in linear time) a decomposition-tree T of G. Now, let us describe the algorithm that proceeds bottom-up from the leaves to the root of T. Every leaf of T corresponds to a subgraph G' with a bounded number of vertices, and therefore, $c_g(G')$ and an optimal strategy for Alice can be computed in time $\mathcal{O}(1)$. For every internal node v (corresponding to a subgraph G' of G) of T, $c_g(G')$ and a corresponding strategy for Alice are computed from what has already been computed for the two subgraphs corresponding to the children of v.

Precisely, let G_1 and G_2 be the two subgraphs of G corresponding to the children of the root of T, and assume by induction that $c_g(G_1)$, $c_g(G_2)$, and optimal strategies for Alice in G_1 and G_2 have been computed in linear time. We now describe how the algorithm proceeds for G, and we set $|V(G)| = n$. There are 4 cases depending on how G is obtained from G_1 and G_2.

1. If $G = G_1 + G_2$, then $c_g(G) = \max\{c_g(G_1), c_g(G_2)\}$ by Lemma 8.3.1. W.l.o.g., $c_g(G) = c_g(G_1)$. By induction, $c_g(G_1)$ and a strategy for Alice have already been computed.
2. If $G = G_1 \oplus G_2$, then $c_g(G) = \lceil \frac{n}{2} \rceil$ by Lemma 8.3.2. Moreover, a corresponding strategy for Alice is also given in the proof of Lemma 8.3.2.
3. Assume that $G = (S, K, R, E)$ is a spider with $G_1 = G[S \cup K]$ and $G_2 = G[R]$. Note that if $|R|$ is odd (even, resp.), then n is odd (even, resp.), as $|S| = |K|$. There are two subcases:

(a) G is an antimatched spider.

Assume that $|K| \geq 3$ since G is a matched spider if $|K| = 2$. Then, $c_g(G) = \lceil \frac{n}{2} \rceil$. Indeed, consider any strategy for Alice where she colours two uncoloured vertices $v_1, v_2 \in K$ in the first two rounds (this is possible since $|K| \geq 3$). Since G is an antimatched spider, for every vertex $v \in S$, at least one of the edges in $\{vv_1, vv_2\}$ is in E. Thus, since K is also a clique and $G[K \cup R] = K \oplus R$, the set $\{v_1, v_2\}$ forms a connected dominating set of G, and we get the result by Lemma 7.1.1.

(b) G is a matched spider.

Let us show that

$$c_g(G) = \begin{cases} \lceil \frac{n}{2} \rceil - \left\lceil \frac{\lfloor \frac{|K|}{2} \rfloor}{2} \right\rceil & \text{if } n \text{ and } \left\lfloor \frac{|K|}{2} \right\rfloor \text{ are odd.} \\ \lceil \frac{n}{2} \rceil - \left\lfloor \frac{\lfloor \frac{|K|}{2} \rfloor}{2} \right\rfloor & \text{otherwise.} \end{cases}$$

Bob's strategy First, we give a strategy for Bob to prove the upper bound on $c_g(G)$ in both cases. Bob first plays exhaustively in K (*i.e.*, until every vertex in K is coloured), then he plays exhaustively in R, then he colours the vertices of S that are matched to red vertices of K, and finally, he colours any remaining uncoloured vertices (the vertices of S that are matched to blue vertices of K). By Bob's strategy, at the end of the game, any red vertex in S that is matched to a blue vertex of K forms a one-vertex connected red component. Let r_S^* be the number of such red vertices. Then, $c_g(G) \leq \lceil \frac{n}{2} \rceil - r_S^*$.

Let us first show that $r_S^* \geq \left\lfloor \frac{\lfloor \frac{|K|}{2} \rfloor}{2} \right\rfloor$. Let b_K be the number of blue vertices in K once all the vertices of K are coloured. Since Bob first exhaustively colours the vertices in K, we have that $b_K \geq \left\lfloor \frac{|K|}{2} \right\rfloor$. Then, while it is possible, Bob colours vertices that are not vertices of S matched to blue vertices in K. Consider the very first point of the game where no such vertex exists (this can occur after a move made by Alice or Bob). Let $r_S \geq 0$ be the number of vertices in S that, at this point, are red and matched to a blue vertex in K. Now, Bob colours the uncoloured vertices of S matched to blue vertices, and thus, Bob colours at most $\left\lceil \frac{b_K - r_S}{2} \right\rceil$ such vertices. Hence, Alice colours at least $\left\lfloor \frac{b_K - r_S}{2} \right\rfloor$ such vertices. We get that $r_S^* \geq r_S + \left\lfloor \frac{b_K - r_S}{2} \right\rfloor \geq \left\lfloor \frac{b_K}{2} \right\rfloor \geq \left\lfloor \frac{\lfloor \frac{|K|}{2} \rfloor}{2} \right\rfloor$.

Now, let us consider the particular case where n and $\left\lfloor \frac{|K|}{2} \right\rfloor$ are odd, and let us refine the above analysis to show that, in this case, $r_S^* \geq \left\lceil \frac{\lfloor \frac{|K|}{2} \rfloor}{2} \right\rceil$. First, if $b_K > \left\lfloor \frac{|K|}{2} \right\rfloor$, then, since $\left\lfloor \frac{|K|}{2} \right\rfloor$ is odd, we get that $\left\lfloor \frac{b_K}{2} \right\rfloor > \left\lfloor \frac{\lfloor \frac{|K|}{2} \rfloor}{2} \right\rfloor$, and so, $\left\lfloor \frac{b_K}{2} \right\rfloor \geq \left\lceil \frac{\lfloor \frac{|K|}{2} \rfloor}{2} \right\rceil$, implying that $r_S^* \geq \left\lceil \frac{\lfloor \frac{|K|}{2} \rfloor}{2} \right\rceil$. Hence, we may assume that $b_K = \left\lfloor \frac{|K|}{2} \right\rfloor$, and so, b_K is odd. Since n is odd, Alice is the last player to colour a vertex in G. Hence, just before Bob colours his first vertex of S matched to a blue vertex in K, there are an even number of such uncoloured vertices remaining. Since b_K is odd, this implies that $r_S \geq 1$. Hence,

$$r_S^* \geq 1 + \left\lfloor \frac{b_K - 1}{2} \right\rfloor = 1 + \left\lfloor \frac{\left\lfloor \frac{|K|}{2} \right\rfloor - 1}{2} \right\rfloor = 1 + \left\lfloor \frac{\left\lfloor \frac{|K|}{2} \right\rfloor}{2} \right\rfloor \geq \left\lceil \frac{\left\lfloor \frac{|K|}{2} \right\rfloor}{2} \right\rceil .$$

Thus, we have proved the upper bound on $c_g(G)$ in both cases.

Alice's strategy Now, we give a strategy for Alice to prove the lower bound on $c_g(G)$ in both cases. Alice first plays exhaustively in K, then she plays exhaustively in R, then she colours the vertices of S that are matched to red vertices of K, and finally, she colours any remaining uncoloured vertices (the vertices of S that are matched to blue vertices of K). Let r_K be the number of red vertices in K once all the vertices of K are coloured. Since Alice first exhaustively colours the vertices in K, we have that $r_K \geq \left\lceil \frac{|K|}{2} \right\rceil$. Let $b_K = |K| - r_K \leq \left\lfloor \frac{|K|}{2} \right\rfloor$ be the number of blue vertices in K once all the vertices of K are coloured. Let u_S be the number of vertices of S that are matched to blue vertices in K. Obviously, $u_S \leq b_K$. Alice's strategy ensures that, at the end of the game, the red vertices induce one connected component X and (if Bob plays optimally) some isolated vertices in S that are matched to blue vertices in K. By Alice's strategy, there are at most $\lceil \frac{u_S}{2} \rceil$ such isolated red vertices. Hence, $|X| \geq \lceil \frac{n}{2} \rceil - \lceil \frac{u_S}{2} \rceil \geq \lceil \frac{n}{2} \rceil - \left\lceil \frac{b_K}{2} \right\rceil$. Thus, $|X| \geq \lceil \frac{n}{2} \rceil - \left\lceil \frac{\left\lfloor \frac{|K|}{2} \right\rfloor}{2} \right\rceil$, which matches the upper bound when n and $\left\lfloor \frac{|K|}{2} \right\rfloor$ are odd. Also, if $\left\lfloor \frac{|K|}{2} \right\rfloor$ is even, then $\left\lceil \frac{\left\lfloor \frac{|K|}{2} \right\rfloor}{2} \right\rceil = \left\lfloor \frac{\left\lfloor \frac{|K|}{2} \right\rfloor}{2} \right\rfloor$, and so, $|X| \geq \lceil \frac{n}{2} \rceil - \left\lfloor \frac{\left\lfloor \frac{|K|}{2} \right\rfloor}{2} \right\rfloor$.

The last case to consider is when n is even. Then, Bob is the last player to colour a vertex. This implies that Alice colours at most $\lfloor \frac{u_S}{2} \rfloor$ vertices of S matched to blue vertices in K. So, $|X| \geq \lceil \frac{n}{2} \rceil - \lfloor \frac{u_S}{2} \rfloor \geq \lceil \frac{n}{2} \rceil - \left\lfloor \frac{b_K}{2} \right\rfloor \geq \lceil \frac{n}{2} \rceil - \left\lfloor \frac{\left\lfloor \frac{|K|}{2} \right\rfloor}{2} \right\rfloor$.

4. Finally, let us assume that $G = (S, K, R, E)$ is a q-pseudo-spider with $G_1 = G[S \cup K]$ (with $|V(G_1)| \leq q$) and $G_2 = G[R]$. By Lemma 8.3.1, we may assume that G is connected.

   First, let us consider the case where $|V(G_2)| \leq 2q$, and so, $|V(G)| \leq 3q$. An exhaustive search allows to compute $c_g(G)$ and a corresponding strategy for Alice in time $\mathcal{O}(1)$. Roughly, the set of all games in G can be described by one rooted tree with maximum degree at most $3q$ and depth $3q$. A classical dynamic-programming algorithm on this execution-tree can be used to compute the result in time $\mathcal{O}(1)$.

   From now on, let us assume that $|V(G_2)| > 2q$. Note that, in this setting, as soon as Alice colours a vertex of G_2 (and she will always be able to do that in the strategies below because $|V(G_2)| > 2q$), all the red vertices of K will belong to the same connected red component (since $G[K \cup R] = K \oplus R$). Moreover, in what follows, Alice will always colour at least $\left\lfloor \frac{|V(G_2)|}{2} \right\rfloor \geq q$ vertices in G_2, connected by a vertex of K, ensuring that the largest connected red component is always this one (the one containing all the red vertices of G_2) since $|V(G_1)| \leq q$.

In what follows, we make use of the following slight variation of the Maker-Breaker Largest Connected Subgraph game. Consider the following game that takes a graph H and $X \subseteq V(H)$ as inputs. The game proceeds as the Maker-Breaker Largest Connected Subgraph game does, *i.e.*, Alice and Bob take turns colouring vertices of G starting with Alice and with all the vertices being initially uncoloured. The difference lies in the objective of Alice. At the end of the game, the score achieved by Alice is the total number of red vertices that belong to the connected red components containing vertices of X. Intuitively, we see all the connected red components with at least one vertex in X as a single connected red component. Let $c_g(H, X)$ be the largest integer k such that Alice has a strategy to ensure a score of at least k with input (H, X), regardless of how Bob plays. Note that, by arguments similar to those near the beginning of this proof, if $|V(H)| = \mathcal{O}(1)$, then $c_g(H, X)$ (and a corresponding strategy for Alice) can be computed in time $\mathcal{O}(1)$ for all $X \subseteq V(H)$.

By the previous remark, $c_g(G_1, K)$ (and a corresponding strategy $\mathcal{S}_a^1$ for Alice) can be computed in time $\mathcal{O}(1)$. By an exhaustive computation in constant time (since $|V(G_1)| = \mathcal{O}(1)$), it is actually possible to consider all the strategies for Alice and Bob, including the ones where they may each skip one of their turns. If (in the variant game with input (G_1, K)) there exists a strategy for Alice guaranteeing her a score of at least $c_g(G_1, K)$, in which she skips one of her turns, and such that, if Bob skips a turn before Alice, then Alice can score at least $c_g(G_1, K) + 1$ without skipping any of her turns, then let $\mathcal{S}_a^2$ be such a strategy for Alice. On the other hand, if (in the variant game with input (G_1, K)) there exists a strategy for Bob guaranteeing that Alice cannot score more than $c_g(G_1, K)$, in which he skips one of his turns, and such that, if Alice skips a turn before Bob, then Bob can guarantee that Alice scores at most $c_g(G_1, K) - 1$ without skipping any of his turns, then let $\mathcal{S}_b^2$ be such a strategy for Bob. Note that, by definition, $\mathcal{S}_a^2$ and $\mathcal{S}_b^2$ cannot both exist simultaneously.

Now, let us consider the following strategy $\mathcal{S}_b$ for Bob. Whenever Alice colours a vertex in G_1, Bob plays in G_1 following a strategy that ensures that Alice scores at most $c_g(G_1, K)$ in the variant game with input (G_1, K). Whenever Alice colours a vertex in G_2, Bob colours any vertex of G_2 (if no such move is possible, Bob colours any arbitrary uncoloured vertex in G). This ensures that the largest connected red component is of order at most $c_g(G_1, K) + \left\lceil \frac{|V(G_2)|}{2} \right\rceil$. That is, $c_g(G) \leq c_g(G_1, K) + \left\lceil \frac{|V(G_2)|}{2} \right\rceil$.

Let us also define the following strategy $\mathcal{S}_a$ for Alice. First, Alice colours the first vertex in G_1 that ensures her a score of at least $c_g(G_1, K)$ in the variant game with input (G_1, K) (following strategy $\mathcal{S}_a^1$). Then, whenever Bob colours a vertex in G_1, Alice colours the vertex of G_1 following her strategy $\mathcal{S}_a^1$ to ensure a score $c_g(G_1, K)$ in the variant game with input (G_1, K). Whenever Bob colours a vertex in G_2, Alice colours any vertex in G_2. If no such move is possible, Alice colours any arbitrary uncoloured vertex. This ensures that the largest connected red component is of order at least $c_g(G_1, K) + \left\lfloor \frac{|V(G_2)|}{2} \right\rfloor$ (recall that, since $|V(G_2)| \geq 2$, Alice colours at least one vertex in G_2). That is, $c_g(G) \geq c_g(G_1, K) + \left\lfloor \frac{|V(G_2)|}{2} \right\rfloor$.

Note that the upper and lower bounds above match when $|V(G_2)|$ is even. Assume now that $|V(G_2)|$ is odd. We distinguish three cases in what follows. In all of the strategies below, the first player to colour a vertex in G_2 will colour at least $\left\lceil \frac{|V(G_2)|}{2} \right\rceil$ vertices in G_2.

— First, let us assume that the strategy $\mathcal{S}_a^2$ for Alice in G_1 defined above exists. In that case, let us define Alice's strategy for G as follows. Alice plays her first turns in G_1 following $\mathcal{S}_a^2$ until she can skip a turn in G_1 (*i.e.*, the first time she can skip a turn in G_1 while still guaranteeing a score of at least $c_g(G_1, K)$ in the variant game with input (G_1, K)).
  — If, in one of these rounds, Bob plays in G_2, then Alice first plays an extra turn in G_1 (following $\mathcal{S}_a^2$ that ensures her a score of at least $c_g(G_1, K) + 1$ in the variant game with input (G_1, K)), and then, each time Bob plays in G_1, she plays in G_1 according to $\mathcal{S}_a^2$ in the variant game with input (G_1, K), and each time Bob plays in G_2, she plays in G_2.
  — Otherwise, Bob also plays in G_1 until Alice can skip a turn in G_1. Then, once she can skip a turn in G_1 according to $\mathcal{S}_a^2$, Alice colours a vertex in G_2. From then, whenever Bob colours a vertex in G_1, she colours a vertex in G_1 following $\mathcal{S}_a^2$. Otherwise, she colours any arbitrary vertex in G_2.

  In both cases, this guarantees Alice a score of at least $c_g(G_1, K) + \left\lceil \frac{|V(G_2)|}{2} \right\rceil$, matching the upper bound.

— Second, let us assume that the strategy $\mathcal{S}_b^2$ for Bob in G_1 defined above exists. Note that $\mathcal{S}_a^2$ does not exist, so Alice cannot skip one turn in G_1 without decreasing her score in the variant game with input (G_1, K). Bob plays his first turns in G_1 following $\mathcal{S}_b^2$ until he can skip a turn in G_1.
  — If, in one of these rounds, Alice plays in G_2, then Bob first plays an extra turn in G_1 (following $\mathcal{S}_b^2$ that ensures him that Alice will score at most $c_g(G_1, K) - 1$ in the variant game with input (G_1, K)). Then, whenever Alice plays in G_1, he continues to follow $\mathcal{S}_b^2$ in the variant game with input (G_1, K), and when Alice plays in G_2, Bob plays in G_2.
  — Otherwise, Alice also plays in G_1 until Bob can skip a turn in G_1. Then, once he can skip a turn in G_1 according to $\mathcal{S}_b^2$, Bob colours a vertex in G_2. From then, whenever Alice colours a vertex in G_1, he colours a vertex in G_1 following $\mathcal{S}_b^2$. Otherwise, he colours any arbitrary vertex in G_2.

  In both cases, this guarantees that Alice's score is at most $c_g(G_1, K) + \left\lfloor \frac{|V(G_2)|}{2} \right\rfloor$, matching the lower bound.

— Finally, if none of the strategies $\mathcal{S}_a^2$ and $\mathcal{S}_b^2$ exist, the result depends on the parity of $|V(G_1)|$. Indeed, if Alice skips one turn in G_1, then Bob can ensure she scores at most $c_g(G_1, K) - 1$ in the variant game with input (G_1, K). On the other hand, if Bob skips one turn in G_1, Alice can score at least $c_g(G_1, K) + 1$ in the variant game with input (G_1, K). For Alice to ensure her upper bound and for Bob to ensure the lower bound, both of them will play in priority in G_1. That is, the first vertex of G_2 is coloured after all the vertices of G_1 have been coloured (and Alice has achieved a score of $c_g(G_1, K)$ in the variant game with input (G_1, K)). If $|V(G_1)|$ is even, Alice is the first player to colour a vertex in G_2, which allows her to score the upper bound $c_g(G_1, K) + \left\lceil \frac{|V(G_2)|}{2} \right\rceil$. Otherwise, Bob is the first player to colour a vertex in G_2, which implies that Alice can score at most the lower bound $c_g(G_1, K) + \left\lfloor \frac{|V(G_2)|}{2} \right\rfloor$.

$\square$

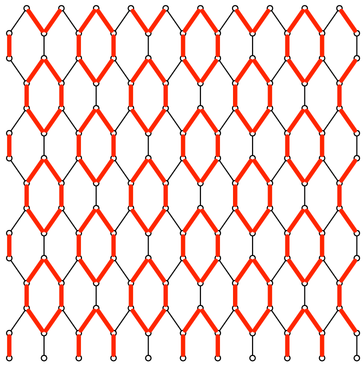


Figure 8.6 – Illustration of the infinite Hexagonal grid G^H_∞ in the proof of Proposition 8.3.7. The connected red subgraphs are vertex-disjoint 6-cycles covering all the vertices of G^H_∞. The black edges induce a matching of G^H_∞.

We close this section by providing an additional way in which Lemma 8.3.1 can be used. Indeed, apart from dealing with graphs with multiple connected components, the proof of Lemma 8.3.1 also implies that for every subgraph H of G, $c_g(H) \leq c_g(G)$. To illustrate the usefulness of this result we consider the case of playing the Maker-Breaker game in *hexagonal grids*.

Proposition 8.3.7. *If G is a finite subgraph of the infinite hexagonal grid, then $c_g(G) \leq 6$.*

Proof. Let G^H_∞ be the infinite hexagonal grid as partially shown in Figure 8.6. By the proof of Lemma 8.3.1, it is sufficient to show that $c_g(G^H_\infty) \leq 6$. Let $(C_i)_{i\in\mathbb{N}}$ be the set of vertex-disjoint subgraphs of G^H_∞ depicted in red in Figure 8.6. Note that, for any $i \in \mathbb{N}$, C_i induces a cycle of order 6 and $(V(C_i))_{i\in\mathbb{N}}$ is a partition of $V(G^H_\infty)$. Furthermore, $M = E(G^H_\infty) \setminus (\bigcup_{i\in\mathbb{N}} E(C_i))$ (black edges in Figure 8.6) is a matching of G^H_∞. Note also that, for any $i \neq j$, every path from a vertex of C_i to a vertex of C_j contains an edge in M (since, for every subgraph C_i, the edges adjacent to a vertex of $V(C_i)$, but not in $E(C_i)$, are by definition in M).

Let us consider the following strategy for Bob. First, note that, for any vertex $v \in V(G)$, there is at most one edge $uv \in M$ incident to v since M is a matching. Thus, each time Alice colours a vertex v, Bob colours the vertex u such that $uv \in M$, if it exists and it is uncoloured, and if not, then he colours any arbitrary uncoloured vertex in G. Let us show that Bob's strategy ensures that Alice cannot create a connected red component of order more than 6. Towards a contradiction, let us assume that Alice creates a connected red component S of order at least 7. Then, there exist $u, v \in S$ and $i \neq j$ such that $u \in V(C_i)$ and $v \in V(C_j)$ (because the C_k's partition the vertex-set of G^H_∞ and each C_k has order 6). As was mentioned above, every path between u and v must contain an edge of M, and so, by Bob's strategy, a vertex of this path has been coloured by Bob, contradicting that u and v belong to the same connected red component. □

Through a tedious case analysis, it might be possible to prove that $c_g(G^H_\infty) = 6$. However, the case of other classic types of grids seems trickier, as indicated by the partial result we have already provided in Proposition 8.1.7.

8.4 Conclusion

In this chapter we treated some “simple” families of graphs. We first determined the outcome of the Scoring game when played on paths and cycles. We then showed that deciding the outcome of either game when played on cographs can be done in linear time. Finally, we provided a linear time algorithm that decides the outcome of the Maker-Breaker game when played on $(q, q-4)$-graphs, a family that generalises cographs.

As illustrated by the cases of paths and cycles, and cographs, there is not a unique way to play the Scoring game. Indeed, Alice and Bob, depending on the graph’s properties, might have several strategical options to choose from. Each such strategy is already interesting by itself, and could thus be subject to a dedicated focus as further work on the topic.

An even more interesting direction is to compare such strategies for the two introduced games. Indeed, Lemma 8.3.1 draws a neat difference between the two versions of the game, as the outcome of the Scoring game in a disconnected graph cannot be established as simply as in the Maker-Breaker game. This is because, in the latter version, Bob does not care about the structure induced by the blue vertices. However, in the Scoring game, there are scenarios in which it is more favourable for Bob to play in a connected component G_2 different from the one G_1 that Alice just played in. This would be like skipping a turn in G_1, but playing an extra turn in G_2 (or playing first in G_2). Thus, to establish a result similar to Lemma 8.3.1 for the Scoring game, one has to deal with the effects of skipping and playing extra turns, as well as Bob playing first, which seems like a tricky, yet interesting, aspect to study. Such a result would also be helpful for extending our linear time algorithm for deciding the outcome of the Scoring game on cographs to $(q, q-4)$-graphs, as we did in Theorem 8.3.6 for the Maker-Breaker game.

Regarding determining c_g for other graph classes, an appealing direction could be to consider standard graph classes such as trees. Theorem 8.1.3 implies that $c_g(P_n) = 2$ for any path P_n of order $n \geq 3$, and we believe that understanding the Maker-Breaker game in larger subclasses of trees such as caterpillars and subdivided stars is feasible, but requires a lot of work to prove, for a not so substantial result. Thus, we think it would be most interesting to study directly the class of trees rather than its subclasses. Other natural graph classes to be investigated are graph products. For instance, we wonder whether $c_g(Q_n)$ can be easily determined for a hypercube Q_n (where, recall, Q_2 is the cycle C_4 of length 4, and, for every $n > 2$, the hypercube Q_n is the Cartesian product $Q_{n-1} \square P_2$ of Q_{n-1} and the path P_2 of order 2). Also, in the upcoming Chapter 9, we will discuss about king’s grids for which we will provide bounds on c_g when there are two rows and m columns.

Regarding Proposition 8.1.7, we would be interested in knowing the precise value of $c_g(P_n \square P_m)$ in general. One issue we ran into is the fact that Alice can play in a non-connected way (meaning that during a turn Alice can colour a vertex that is not adjacent to a red vertex) and it is not clear how Bob should anticipate to prevent connected red components to merge later on. This is the first indication that a connected variant of the Maker-Breaker game, where Alice is only allowed to colour neighbours of red vertices after the first turn, is an interesting game to be studied. We comment further on this variant in the conclusion of the upcoming Chapter 9 as some arguments presented in that chapter, in particular in Section 9.3, will give us a better understanding of the behaviour of such a variant.

CHAPTER 9

A-perfect graphs and regularity

In this chapter we focus on the Maker-Breaker games, and specifically on the question of identifying A-perfect graphs. We show that there exist arbitrarily large A-perfect d-regular graphs for any $d \geq 4$, but, surprisingly, any 3-regular A-perfect graph is of order strictly less than 18. Moreover, we give sufficient conditions, in terms of the number of edges or the maximum and minimum degrees, for a graph to be A-perfect.

This chapter focuses on the Maker-Breaker game. In Section 9.1 we provide some additional motivation for introducing this game. We also provide lower and upper bounds on the parameter c_g. Then, looking for graphs that verify the previously given bounds, we look into connected regular graphs that are A-perfect. In Section 9.2.1 we provide constructions for arbitrarily large connected d-regular graphs that are A-perfect, for $d \geq 4$. Then, in Section 9.2.2, we show that any 3-regular A-perfect graph is of order strictly less than 18. We then move on to Section 9.3, where we provide sufficient conditions for a graph to be A-perfect, depending on the degrees or the size of the graph.

9.1 Preliminaries

Allow us to better describe our motivation for introducing the Maker-Breaker game, with regards to the Scoring game. In the latter game, for certain subgraphs of some graphs, it may not be interesting for Bob to try to increase his score in those subgraphs, but rather just to limit Alice's score in them. In particular, this can be the case in graphs that are not connected (as we exhibited for the case of cographs in Chapter 8). Understanding just how much Bob can limit Alice's score in these subgraphs is equivalent to playing the Maker-Breaker game in them. Another motivation is to understand the properties of graphs in which Alice can ensure a single connected red component at the end of the Scoring game (especially since Alice wins in these graphs if they have odd order). This leads us to consider A-perfect graphs in the context of the Maker-Breaker game, which are the graphs G for which $c_g(G) = \lceil |V(G)|/2 \rceil$.

To avoid any confusion, let us clarify that in this chapter we are only interested by A-perfect graphs for the Maker-Breaker game. Note that the proofs of Theorem 7.3.3 and Corollary 7.3.4 imply that:

Corollary 9.1.1. *Deciding if a graph G is A-perfect is PSPACE-complete, even if G is a bipartite graph with diameter 4 or a split graph.*

We close this section with a first lemma that treats the general upper and lower bounds for the parameter $c_g(G)$:

Lemma 9.1.2. *For every graph G,* $\left\lfloor \frac{\Delta(G)}{2} \right\rfloor + 1 \leq c_g(G) \leq \left\lceil \frac{|V(G)|}{2} \right\rceil$.

Proof. The right-hand side of the inequality follows from the fact that Alice always colours exactly $\left\lceil \frac{|V(G)|}{2} \right\rceil$ vertices. We now give a strategy for Alice that ensures a score of at least $\left\lfloor \frac{\Delta(G)}{2} \right\rfloor + 1$, to prove the left-hand side of the inequality. In the first round, Alice colours a vertex v with degree $\Delta(G)$. Then, in each of the next rounds, if possible, Alice colours an uncoloured neighbour of v. Once the game ends, by the strategy above, Alice must have coloured v and at least half of its neighbours, and the result follows. □

Both bounds in Lemma 9.1.2 can be reached for arbitrarily large graphs. For example, there exist arbitrarily large connected graphs that are A-perfect, since every graph with a universal vertex is A-perfect. Regarding the lower bound, the graph G that is the disjoint union of m copies of the complete graph K_{d+1} (for any $d \in \mathbb{N}$) is d-regular, and $c_g(G) = \left\lfloor \frac{d}{2} \right\rfloor + 1$, while G gets more and more distant from being A-perfect as m increases.

This last remark makes us wonder about the tightness of the bounds in Lemma 9.1.2 for arbitrarily large regular connected graphs. In the following sections we establish that there exist arbitrarily large connected d-regular graphs G, with $d \geq 3$, for which $c_g(G)$ is close to the lower bound (Lemma 9.2.1), while, for every $d \geq 4$, there exist arbitrarily large connected d-regular graphs G that are A-perfect (Lemma 9.2.2). Surprisingly, the latter result does not hold for every $d \geq 3$, as we prove that any sufficiently large cubic graph is not A-perfect (Theorem 9.2.4).

9.2 A-perfect regular graphs

Before stating our results on $c_g(G)$ where G is a regular graph, note that the case of 2-regular graphs, *i.e.*, cycles, has already been dealt with through Proposition 8.1.6. Thus, in what follows we consider d-regular graphs, for $d \geq 3$.

9.2.1 Regular graphs reaching the bounds of c_g

We now show that the lower bound in Lemma 9.1.2 is almost tight for arbitrarily large connected d-regular graphs, for every $d \geq 3$.

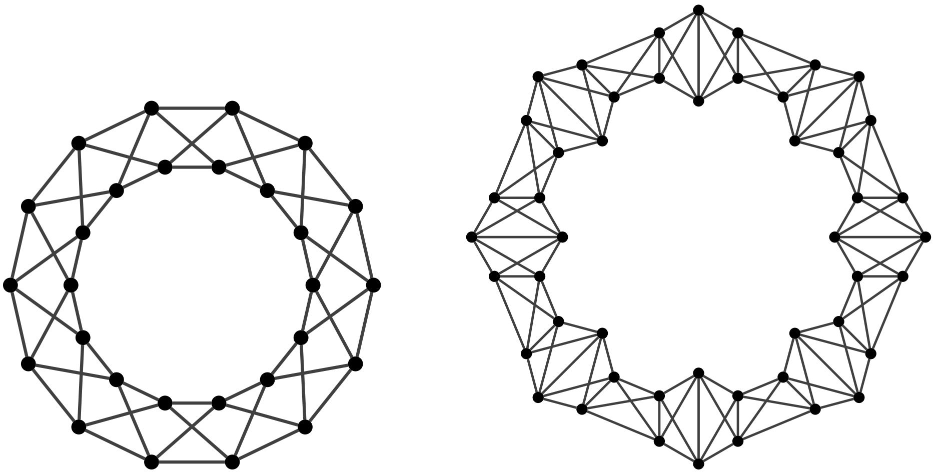

(a) $d = 4$ (b) $d = 5$

Figure 9.1 – Examples of d-regular A-perfect graphs constructed in the proof of Lemma 9.2.2.

Lemma 9.2.1. *For every $d \geq 3$, there exist arbitrarily large connected d-regular graphs G such that $c_g(G) \leq \left\lceil \frac{d+3}{2} \right\rceil$.*

Proof. Let G be the graph constructed as follows. Start from $N \geq 2$ disjoint copies $H_0, \dots, H_{N-1}$ of the complete graph on $d+1$ vertices. Now, for every $i \in \{0, \dots, N-1\}$, remove the edge $u_i v_i$, where u_i and v_i are any two distinct vertices of H_i. Finally, add the edge $v_i u_{i+1}$ for every $i \in \{0, \dots, N-1\}$ (where, here and further, operations are understood modulo N). Note that the resulting graph G is d-regular, and, free to consider large values of N, can be as large as desired. For every $i \in \{0, \dots, N-1\}$, every vertex of H_i that is different from u_i and v_i is said to be *internal* (to H_i). Since $d \geq 3$, every H_i has at least two internal vertices.

We give a strategy for Bob that ensures that Alice's score in G is at most $\left\lceil \frac{d+3}{2} \right\rceil$. In each round, if the last vertex coloured by Alice is

- some vertex u_i, then Bob colours v_{i-1};
- some vertex v_i, then Bob colours u_{i+1};
- a vertex internal to some H_i, then Bob colours an uncoloured vertex internal to the same H_i.

By this strategy, once the game ends, every connected red component must be completely contained inside some H_i. This is because this strategy guarantees that any two vertices v_i and u_{i+1} end up coloured either by different players, or by Bob only. It thus follows that the largest connected red component contains, in the worst-case scenario, some u_i, v_i, and half of the other vertices of H_i. In other words, the largest connected red component is of order at most $\left\lceil \frac{d+3}{2} \right\rceil$. $\square$

Regarding the upper bound from Lemma 9.1.2 in the context of arbitrarily large connected d-regular graphs, we prove the following:

Lemma 9.2.2. *For every $d \geq 4$, there exist arbitrarily large d-regular A-perfect graphs.*

Proof. Let $N > 2$, and let $d \geq 4$ be fixed. To prove the claim, we construct a d-regular graph G, whose order is a function of N, such that $c_g(G) = \left\lceil \frac{|V(G)|}{2} \right\rceil$. We give two possible constructions for G, depending on whether $d = 4$ or $d \geq 5$ (see Figure 9.1 for an illustration of both constructions).

— For the case $d = 4$, G is the 4-regular graph having two vertices u_1^i and u_2^i for every $i \in \{0, \dots, N-1\}$, and the four edges $u_1^i u_1^{i+1}$, $u_1^i u_2^{i+1}$, $u_2^i u_1^{i+1}$, $u_2^i u_2^{i+1}$ for every $i \in \{0, \dots, N-1\}$ (where, here and further, operations over the superscripts are modulo N).
To prove that G is A-perfect, we give a strategy for Alice that ensures that, at the end of the game in G, the red subgraph is connected. In the first round, Alice colours u_1^0. Then, in the subsequent rounds, if the last vertex Bob coloured is u_1^j (u_2^j, resp.) for some $j \in \{1, \dots, N-1\}$, Alice responds by colouring u_2^j (u_1^j, resp.). Otherwise, Alice colours any arbitrary uncoloured vertex. By Alice's strategy, at the end of the game, for every $0 \leq i \leq N-1$, exactly one of u_1^i and u_2^i is red, and thus, the red subgraph is connected, and G is A-perfect.

— We now consider the case where $d \geq 5$. Here, G is constructed as follows. Start from N disjoint copies $H_0, \dots, H_{N-1}$ of the complete graph on $d+1$ vertices, where, for every $i \in \{0, \dots, N-1\}$, we denote by $v_1^i, \dots, v_{d+1}^i$ the vertices of H_i. For every $i \in \{0, \dots, N-1\}$, we remove the edges $v_1^i v_3^i$, $v_1^i v_4^i$, $v_2^i v_3^i$ and $v_2^i v_4^i$ from H_i. To finish the construction of G and make it d-regular, we then join the H_i's by adding the edges $v_3^i v_1^{i+1}$, $v_3^i v_2^{i+1}$, $v_4^i v_1^{i+1}$, and $v_4^i v_2^{i+1}$ for every $i \in \{0, \dots, N-1\}$ (again, operations are understood modulo N).
To prove that G is A-perfect, we give a strategy for Alice that ensures her a score of $\lceil |V(G)|/2 \rceil$. In the first round, Alice colours any vertex. In each of the subsequent rounds, if the last vertex Bob coloured is
  — in some pair $\{v_1^i, v_2^i\}$ or $\{v_3^i, v_4^i\}$, then Alice colours the other vertex in that pair;
  — some vertex v_j^i with $5 \leq j \leq d+1$, then Alice colours another vertex v_ℓ^i with $5 \leq \ell \leq d+1$ and $j \neq \ell$.

  Whenever Alice cannot follow the strategy above, she colours any arbitrary uncoloured vertex. By Alice's strategy, at the end of the game, for every $i \in \{0, \dots, N-1\}$, at least one vertex in $\{v_1^i, v_2^i\}$ is red, at least one vertex in $\{v_3^i, v_4^i\}$ is red, and at least one vertex in $\{v_5^i, \dots, v_{d+1}^i\}$ is red. These vertices form a connected dominating set of G, from which we deduce that $c_g(G) = \lceil |V(G)|/2 \rceil$, by Lemma 7.1.1. Thus, G is A-perfect.

□

9.2.2 The peculiar case of cubic graphs

As mentioned earlier, the bound on d in the statement of Lemma 9.2.2 cannot be lowered, as, surprisingly, we prove that A-perfect cubic graphs have bounded order.

This can actually be established through previous results on the existence of particular cuts in sufficiently large connected cubic graphs, such as ones from [99] relying on the following terminology.

A *supercycle* is a connected graph with minimum degree at least 2 where not all vertices are of degree 2. For a graph G, a matching M is said *suitable* if $G - M$ consists of exactly two connected components, each of which is a supercycle. Note that if G is cubic, then, in $G - M$, every vertex incident to an edge of M has degree precisely 2, while, by definition of a supercycle, each of the two connected components contains a degree-3 vertex. The auhtors of [99] proved the following result on the existence of suitable matchings in sufficiently large connected cubic graphs.

Theorem 9.2.3 (Corollary 1 of [99])**.** *Every connected cubic graph with order at least* 18 *admits a suitable matching.*

We are now ready to prove the aforementioned result on cubic graphs.

Theorem 9.2.4. *Every A-perfect cubic graph has order at most* 16.

Proof. Let G be an A-perfect cubic graph. We can assume that G is connected. Indeed, if G contains at least two connected components, then, assuming Alice starts the game by colouring a vertex in some connected component C, Bob can guarantee the red subgraph contains at least two connected components by the end of the game by just colouring uncoloured vertices of C as long as possible, and then colouring uncoloured vertices arbitrarily until the game ends. By this strategy, and because all connected components of G have order more than 2 due to G being cubic, Alice indeed colours vertices from at least two distinct connected components of G, and the red subgraph is thus not connected once the game ends.

Towards proving the claim, assume now that G has order at least 18. Then, by Theorem 9.2.3, G admits a suitable matching M. As mentioned earlier, $G - M$ consists of exactly two connected components C_1 and C_2, in each of which all vertices incident to an edge of M have degree exactly 2 while the other vertices (there is at least one such) have degree exactly 3. Since, by the handshaking lemma, in every graph the number of odd-degree vertices is even, we deduce that, in each of C_1 and C_2, there are actually at least two degree-3 vertices. In what follows, the degree-2 vertices of the C_i's are called *interior vertices*, while their other (degree-3) vertices are called *exterior vertices*.

Consider now the strategy for Bob where, each turn during a game on G, he answers to Alice's moves as follows:

- if Alice colours an interior vertex incident to an edge $e \in M$, then Bob colours the second interior vertex incident to e;
- if Alice colours an exterior vertex v, then Bob plays as follows:
  - if v is the first exterior vertex coloured by Alice during the whole game, then, denoting, for the rest of the game, by C^* the one of C_1 and C_2 that contains v, Bob colours any uncoloured exterior vertex of C^* (one such exists, since C^* contains at least two exterior vertices);
  - if v is not the first exterior vertex that Alice colours, then C^* was defined during an earlier turn, and Bob colours any uncoloured exterior vertex of C^*. If C^* does not contain any such uncoloured vertex, then Bob colours any uncoloured vertex of G instead.

Note that Bob can clearly follow the above strategy from start to end. Note also that once the game ends, the two incident vertices of every edge of M are coloured with distinct colours. Furthermore, since C_1 and C_2 have at least two exterior vertices each, each C_i must contain an exterior vertex u_i coloured red. From all these arguments we deduce that the red subgraph cannot contain a path joining u_1 and u_2, and thus that the red subgraph is not connected. Thus, an A-perfect cubic graph must have order strictly less than 18. □

9.3 Sufficient conditions for graphs to be A-perfect

We have already seen a few conditions for graphs to meet the upper bound in Lemma 9.1.2, *i.e.*, to be A-perfect. In this section, we give two more such sufficient conditions, one is based on particular degree conditions, while the other is based on the number of edges.

9.3.1 Graphs with large degrees

The next result gives a sufficient condition, in terms of minimum degree and maximum degree, for a graph to be A-perfect.

Theorem 9.3.1. *If G is a connected graph with $\Delta(G) + \delta(G) \geq |V(G)|$, then G is A-perfect.*

Proof. We give a strategy for Alice ensuring that, at the end of the game, the red subgraph is connected, which implies that G is A-perfect. Let u be any vertex of degree $\Delta(G)$. In the first round, Alice colours u. For every $i \geq 1$, let C_i be the connected component of red vertices at the end of the i^{th} round (we will show that the red vertices always induce a connected subgraph, and so, C_i is well-defined). Let $R_i = V(G) \setminus N[C_i]$, *i.e.*, R_i is the set of (non-red) vertices not dominated by a red vertex at the end of the i^{th} round, and let R_i^U be the subset of uncoloured vertices in R_i at the end of the i^{th} round. Note that $C_1 = \{u\}$ is connected and that

$$|R_1^U| \leq |R_1| = |V(G)| - |N[C_1]| = |V(G)| - \Delta(G) - 1 \leq \delta(G) - 1.$$

Let us show by induction on $i \geq 1$ that, at the end of the i^{th} round, C_i is connected and either $R_i^U = \emptyset$ (in which case we are done) or $|R_i^U| \leq \delta(G) - i$. By the above paragraph, the induction hypothesis holds for $i = 1$. Let $i \geq 1$ and let us assume that the induction hypothesis holds for i. We show it still holds for $i + 1$.

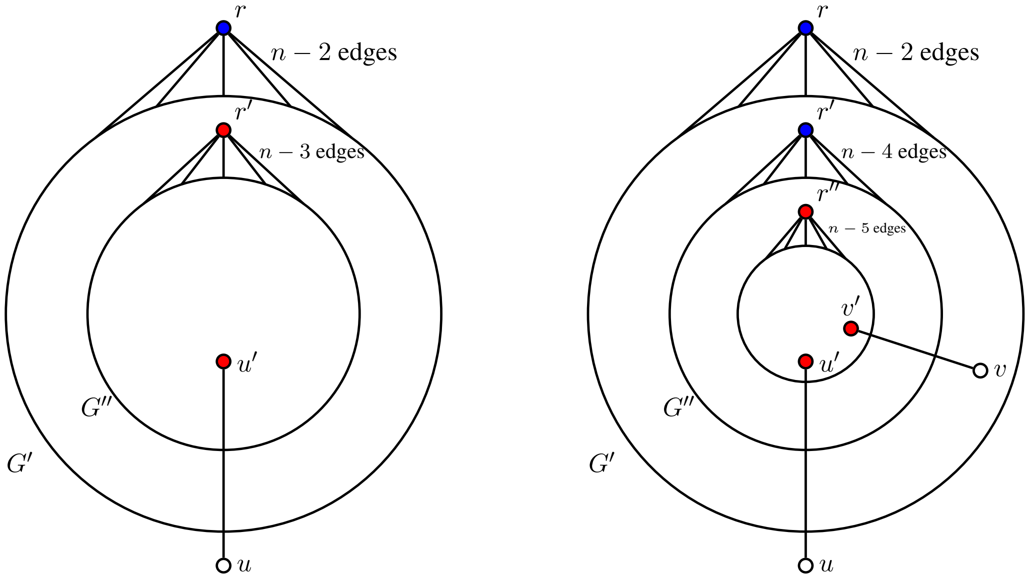
If $R_i^U = \emptyset$, then C_i is a connected red dominating set of the subgraph of G induced by the vertices of C_i and the remaining uncoloured vertices of G. From now on, Alice may colour any uncoloured vertex, and the induction hypothesis clearly holds for $i + 1$. In particular, the set of red vertices induces a connected subgraph at the end of the game, proving the result.

Otherwise, let $v \in R_i^U$. Since v has at least $\delta(G)$ neighbours (none of which are red since $N[R_i] \cap C_i = \emptyset$) and Bob has coloured i vertices, v has at least $\delta(G) - i$ uncoloured neighbours, and $\delta(G) - i > 0$ since $R_i^U \neq \emptyset$ and $|R_i^U| \leq \delta(G) - i$. Moreover, $|R_i^U \setminus \{v\}| < \delta(G) - i$, so v has at least one uncoloured neighbour w not in R_i, which implies that $w \in N(R_i) = N(C_i)$. In the $(i + 1)^{th}$ round, Alice colours w. Then, $C_{i+1} = C_i \cup \{w\}$ is clearly connected, and $R_{i+1}^U \subseteq R_{i+1} \subseteq R_i \setminus \{v\}$ (since $v \in N(C_{i+1})$), and hence, $|R_{i+1}^U| \leq |R_{i+1}| \leq |R_i| - 1 \leq \delta(G) - (i + 1)$. □

We note that the bound in the statement of Theorem 9.3.1 is sharp, in the sense that there exists a graph G with $\Delta(G) + \delta(G) = |V(G)| - 1$ that is not A-perfect. For example, consider the graph G consisting of two complete graphs on $d \geq 3$ vertices joined by a single edge e. Then, $\Delta(G) = d$, $\delta(G) = d - 1$, $|V(G)| = 2d$, and thus, $\Delta(G) + \delta(G) = 2d - 1 = |V(G)| - 1$. However, Bob can guarantee that Alice achieves a score of about $|V(G)|/4$, by colouring an uncoloured vertex incident to e in the first round, and then, in each subsequent round, colouring an uncoloured vertex in the same clique that Alice just coloured a vertex in. Thus, G is not A-perfect.

9.3.2 Graphs with large size

The next result shows that if G has sufficiently many edges, then G is A-perfect.



(a) The state of the game in G after Alice's first two turns in Case 1.(a), where $G'' = G'[N(r')]$.

(b) The state of the game in G after Alice's first three turns in Case 1.(b)i.

Figure 9.2 – Cases 1.(a) and 1.(b)i. in the proof of Theorem 9.3.2.

Theorem 9.3.2. *If G is a connected graph with $|E(G)| > \frac{(|V(G)|-2)(|V(G)|-3)}{2} + 2$, then G is A-perfect.*

Proof. Set $n = |V(G)|$, $m = |E(G)|$, and

$$x = \frac{(n-2)(n-3)}{2} + 2 = \frac{n^2 - 5n + 10}{2}.$$

Note first that $\Delta(G) \geq n - 4$. Indeed, if we had $\Delta(G) \leq n - 5$, then we would deduce that $m \leq \frac{n(n-5)}{2} < x$, which contradicts that $m > x$. Furthermore, if $\Delta(G) = n - 4$, then $\delta(G) \geq 7$. Indeed, if there is a degree-6 vertex, then we have a contradiction since

$$m \leq \frac{(n-1)(n-4) + 6}{2} = \frac{n^2 - 5n + 10}{2} = x.$$

Thus, if $\Delta(G) = n - 4$, then G is A-perfect by Theorem 9.3.1 since $\delta(G) \geq 7$. Lastly, if $\Delta(G) = n - 1$, then $\delta(G) \geq 1$, and thus, G is A-perfect by Theorem 9.3.1. Hence, in what follows, we assume that $n - 3 \leq \Delta(G) \leq n - 2$. We give a strategy for Alice that allows her to colour the vertices of a connected dominating set of G within the first four rounds, and so, by Lemma 7.1.1, G is A-perfect. We treat the two possible values for $\Delta(G)$ independently.

1. $\Delta(G) = n - 2$.

   Let $r \in V(G)$ be such that $d(r) = n - 2$, and let $G' = G[N(r)]$. Then, $|V(G')| = n - 2$. Since $d(r) = n - 2$, there is exactly one additional vertex $u \in V(G) \setminus V(G')$ ($u \neq r$). If $d(u) \geq 2$, then Alice colours r in the first round, and then, in the second round, she colours a neighbour of u (this is possible since $d(u) \geq 2$), and these vertices form a connected dominating set of G. Thus, we may assume that $d(u) = 1$, and let $N(u) = \{u'\}$. We have that $\Delta(G') \geq n-4$. Indeed, if $\Delta(G') \leq n-5$, then we have a contradiction since

$$m \leq \frac{(n-2)(n-5)}{2} + n - 2 + 1 = \frac{n^2-5n+8}{2} < x.$$

We distinguish the following subcases:

(a) $\Delta(G') = n-3$ (see Figure 9.2(a) for an illustration).
Let $r' \in V(G')$ be such that $d_{G'}(r') = n-3$. Alice's strategy is as follows. She starts by colouring u'. Now, if Bob does not colour r, then Alice continues by colouring r, at which point she has coloured the vertices of the connected dominating set $\{u', r\}$ of G. So, we may assume that Bob colours r in the first round. In the second round, Alice colours r'. Observe that $\{u', r'\}$ also forms a connected dominating set of G since $d_{G'}(r') = n-3$, and thus, $u'r' \in E(G)$.

(b) $\Delta(G') = n-4$.
Let $r' \in V(G')$ be such that $d_{G'}(r') = n-4$, and let $G'' = G'[N(r')]$. We distinguish cases according to whether $u' \in V(G'')$ or not.

i. $u' \in V(G'')$.
Since $d_{G'}(r') = n-4$, there is exactly one additional vertex $v \in V(G') \setminus V(G'')$ ($v \neq r'$). Note that $d_{G'}(v) \geq 1$ because if $d_{G'}(v) = 0$, *i.e.*, $N(v) = r$, then we have a contradiction since

$$m \leq \frac{(n-3)(n-4)}{2} + n - 2 + 1 = \frac{n^2-5n+10}{2} = x.$$

If $d_{G'}(v) \geq 2$, then Alice's strategy is as follows. She starts by colouring u'. As before, Bob is forced to colour r in the first round. In the second round, Alice colours r'. In the third round, if $u' \notin N(v)$, then Alice colours a neighbour $v' \in V(G')$ (this is possible since $d_{G'}(v) \geq 2$). After three rounds, Alice's vertices form a connected dominating set of G.
Assume now that $d_{G'}(v) = 1$, and let $N_{G'}(v) = \{v'\}$ (see Figure 9.2(b) for an illustration). Then, $\Delta(G'') = n-5$. Indeed, if $\Delta(G'') \leq n-6$ (and so, $n \geq 6$), then we have a contradiction since

$$m \leq \frac{(n-4)(n-6)}{2} + n - 4 + n - 2 + 1 + 1 = \frac{n^2-6n+16}{2} \leq x.$$

Let $r'' \in V(G'')$ be such that $d_{G''}(r'') = n-5$, and observe that $v' \in N(r'')$. Alice's strategy is as follows. She starts by colouring u', forcing Bob to colour r. Then, she colours v' forcing Bob to colour r' (similarly to earlier, if Bob does not colour r', then Alice colours r', and thus, has coloured the vertices of a connected dominating set of G). Finally, Alice colours r''. Observe that the vertices u', v', and r'' form a connected dominating set of G.

ii. $u' \notin V(G'')$.
Observe that u' is the only vertex of G' that is not a neighbour of r', and that $d(u') \geq 3$. Indeed, if $d(u') \leq 2$, then we have a contradiction since

$$m \leq \frac{(n-3)(n-4)}{2} + n - 2 + 1 = \frac{n^2-5n+10}{2} = x.$$

Thus, there is at least one edge $u'u''$ with $u'' \in V(G'')$. If $d(u') \geq 4$, then Alice's strategy is as follows. She starts by colouring u', forcing Bob to colour r. Then, she colours r', and, in the third round, she colours one of the remaining uncoloured neighbours of u' in G'' (which exists since $d(u') \geq 4$). These three vertices form a connected dominating set of G.

Otherwise, $d(u') = 3$, and, as in Case 1.(b)i, there exists $r'' \in V(G'')$ such that $d_{G''}(r'') = n - 5$. Alice's strategy is as follows. She starts by colouring u', forcing Bob to colour r. Then, she colours u'', forcing Bob to colour r'. Finally, Alice colours r''. Note that u', u'', and r'' form a connected dominating set of G.

2. $\Delta(G) = n - 3$.

   Observe that G cannot contain two vertices u, v such that $d(u) + d(v) \leq 5$. Indeed, if there are two such vertices, then we have a contradiction since $m \leq \frac{(n-2)(n-3)+5}{2}$, but this is not an integer since $(n-2)(n-3) + 5$ is odd, and thus,

   $$m \leq \frac{(n-2)(n-3)+5-1}{2} = \frac{n^2 - 5n + 10}{2} = x.$$

   Let r be a vertex of G such that $d(r) = n - 3$, and let $G' = G[N(r)]$. Since $d(r) = n - 3$, there are exactly two additional vertices $u, v \in V(G) \setminus V(G')$ ($u, v \neq r$). We distinguish cases according to the degrees of u and v, and note that $d(u) + d(v) \geq 6$. In what follows, when we say that Alice colours a vertex if needed, it means that if it is not necessary (in the sense that such a vertex has already been coloured), then she either colours the vertex she is supposed to colour in the next round, or she colours any arbitrary uncoloured vertex in that round.

   (a) $d(u), d(v) \geq 3$.

   Alice's strategy is as follows. She starts by colouring r. In the second round, she colours an uncoloured vertex in $N(v)$ in G' (this is possible since $d(v) \geq 3$). In the third round, if needed, *i.e.*, if Alice has not coloured a vertex in $N(u)$ yet, Alice colours an uncoloured vertex in $N(u)$ in G' if possible, and if not, then $uv \in E(G)$ and Bob coloured $N(u) \setminus \{v\}$ in the first two rounds, and so, she colours v. Then, by the end of the third round, Alice has coloured r, at least one vertex in $N(v)$ in G', and at least one vertex in $N(u)$, and these vertices form a connected dominating set of G.

   (b) $d(u) = 2$ and $d(v) \geq 4$.

   Alice's strategy is as follows. She starts by colouring r. In the second round, she colours an uncoloured vertex in $N(u)$ in G' if possible, and if not, then $uv \in E(G)$ and Bob coloured $N(u) \setminus \{u\}$ in the first round, and so, she colours v. In the third round, Alice colours an uncoloured vertex in $N(v)$ in G' (this is possible since $d(v) \geq 4$). Then, by the end of the third round, Alice has coloured r, at least one vertex in $N(v)$ in G', and at least one vertex in $N(u)$, and these vertices form a connected dominating set of G.

   (c) $d(u) = 1$ and $d(v) \geq 5$.

   Let $u' \in N(u)$ be a fixed neighbour of u in $N(u)$. In this case, there exists at least one vertex $r' \in G'$ with $d_{G'}(r') \geq n - 5$, as otherwise, we have a contradiction since

   $$m \leq \frac{(n-3)(n-6)}{2} + 2(n-3) + 1 = \frac{n^2 - 5n + 8}{2} < x.$$

Note that v has at least 4 neighbours in G' since $d(v) \geq 5$ and $rv \notin E(G)$. We distinguish the following subcases:

i. $\Delta(G') = n - 4$.
Let $r' \in V(G')$ be such that $d_{G'}(r') = n - 4$, then Alice's strategy is as follows. She starts by colouring u' (it may be that $u' = v$). If Bob colours a vertex in $\{r, r'\}$ (a neighbour $v' \in V(G')$ of v, resp.) in the first round, then, in the second round, Alice colours the other vertex in $\{r, r'\}$ (another neighbour $v^* \in V(G')$ of v, resp.). If Alice coloured a vertex in $\{r, r'\}$ (v^*, resp.) in the second round, then she colours a vertex in $\{v', v^*\}$ ($\{r, r'\}$, resp.) in the third round. After three rounds, Alice's vertices form a connected dominating set of G.

ii. $\Delta(G') = n - 5$.
Let $r' \in V(G')$ be such that $d_{G'}(r') = n - 5$, and let $G'' = G'[N(r')]$. We distinguish cases according to whether $u' \in V(G'')$ or not.

A. $u' \in V(G'')$.
As $u' \in V(G'')$, $uv \notin E(G)$. Since $d(r') = n - 5$, there is exactly one additional vertex $w \in V(G') \setminus V(G'')$ ($w \neq r'$). Note that $d_{G'}(w) \geq 1$ because if $d_{G'}(w) = 0$, *i.e.*, $N(w) = r$, then we have a contradiction since

$$m \leq \frac{(n-4)(n-5)}{2} + 2(n-3) + 1 = \frac{n^2 - 5n + 10}{2} = x.$$

If $d_{G'}(w) \geq 2$, then Alice's strategy is as follows. She starts by colouring u'. As before, Bob is forced to colour r in the first round. Indeed, if he does not, then Alice will colour r in the second round, and then she will colour an uncoloured neighbour $v' \in V(G')$ of v in the third round (this is possible since $d(v) \geq 5$), and her vertices form a connected dominating set of G. In the second round, Alice colours r'. In the third round, if needed, *i.e.*, if $u' \notin N(w)$, Alice colours an uncoloured neighbour $w' \in V(G')$ of w (this is possible since $d_{G'}(w) \geq 2$). In the fourth round, Alice colours an uncoloured neighbour $v' \in V(G')$ of v (this is possible since v has at least 5 neighbours in G' as $uv \notin E(G)$ and $d(v) \geq 5$). At the end of the fourth round, Alice's vertices form a connected dominating set of G.
Assume now that $d_{G'}(w) = 1$, and let $N_{G'}(w) = \{w'\}$. Then, $\Delta(G'') = n - 6$. Indeed, if $\Delta(G'') \leq n - 7$ (and so, $n \geq 7$), then we have a contradiction since

$$m \leq \frac{(n-5)(n-7)}{2} + n - 5 + 2(n-3) + 1 + 1 = \frac{n^2 - 6n + 17}{2} \leq x.$$

Let $r'' \in V(G'')$ be such that $d_{G''}(r'') = n - 6$, and observe that $w' \in N(r'')$. Alice's strategy is as follows. She starts by colouring u'. As before (when $d_{G'}(w) \geq 2$), Bob is forced to colour r in the first round. In the second round, Alice colours w'. Analogously to why Bob was forced to colour r in the first round, Bob is forced to colour r' in the second round. In the third round, Alice colours r''. In the fourth round, Alice colours an uncoloured neighbour $v' \in V(G')$ of v (this is possible since v has at least 5 neighbours in G'). At the end of the fourth round, Alice's vertices form a connected dominating set of G.

B. $u' \notin V(G'')$.

First, assume that $uv \notin E(G)$. Then, u' is the only vertex of G' that is not a neighbour of r', and $d(u') \geq 3$. Indeed, if $d(u') \leq 2$, then we have a contradiction since

$$m \leq \frac{(n-4)(n-5)}{2} + 2(n-3) + 1 = \frac{n^2 - 5n + 10}{2} = x.$$

Thus, there is at least one edge $u'u''$ with $u'' \in V(G'') \cup \{v\}$. If $d(u') \geq 4$, then Alice's strategy is as follows. She starts by colouring u', forcing Bob to colour r, as before. Then, she colours r', and in the third round, she colours one of the remaining uncoloured neighbours of u' in G'' (which exists since $d(u') \geq 4$). In the fourth round, Alice colours an uncoloured neighbour $v' \in V(G')$ of v (this is possible since v has at least 5 neighbours in G'). At the end of the fourth round, Alice's vertices form a connected dominating set of G.

Otherwise, $d(u') = 3$, and, as in Case 2.(c)iiA, there exists $r'' \in V(G'')$ such that $d_{G''}(r'') = n - 6$. Let $r'' \in V(G'')$ be such that $d_{G''}(r'') = n - 6$. Alice's strategy is as follows. She starts by colouring u', forcing Bob to colour r, as before. Then, she colours u'', forcing Bob to colour r', as before. In the third round, Alice colours r''. In the fourth round, Alice colours an uncoloured neighbour $v' \in V(G')$ of v (this is possible since v has at least 5 neighbours in G'). At the end of the fourth round, Alice's vertices form a connected dominating set of G.

Now, assume that $uv \in E(G)$. Then, $u' = v$ and there is exactly one additional vertex $w \in V(G') \setminus V(G'')$ ($w \neq r'$). Note that $d_{G'}(w) \geq 2$ because if $d_{G'}(w) = 1$, then we have a contradiction since

$$m \leq \frac{(n-4)(n-5)}{2} + 2(n-3) + 1 = \frac{n^2 - 5n + 10}{2} = x.$$

Alice's strategy is as follows. She starts by colouring $u' = v$, forcing Bob to colour r, as before. In the second round, she colours r'. In the third round, Alice colours an uncoloured neighbour $w' \in V(G')$ of w (this is possible since $d_{G'}(w) \geq 2$). In the fourth round, if needed, *i.e.*, if Alice has not yet coloured a vertex in $N(v)$ that is not u, Alice colours an uncoloured neighbour $v' \in V(G')$ of v (this is possible since $d(v) \geq 5$). At the end of the fourth round, Alice's vertices form a connected dominating set of G.

□

We note that the bound in the statement of Theorem 9.3.2 is sharp, in the sense that there exists a graph G with $\frac{(|V(G)|-2)(|V(G)|-3)}{2} + 2$ edges that is not A-perfect. For example, consider, as G, any graph obtained from a complete graph on an odd number $N \geq 3$ of vertices, by taking any of its vertices u, and attaching at u a pending path (u, v, w) of length 2. Note that $|V(G)| = N + 2$ and that

$$|E(G)| = \frac{N(N-1)}{2} + 2 = \frac{(|V(G)|-2)(|V(G)|-3)}{2} + 2.$$

Now, to see that G is not A-perfect, consider the following strategy for Bob. Bob colours a vertex in $\{u, v\}$ in the first round, and then, in each of the subsequent rounds, he colours any uncoloured vertex different from w. Since $|V(G)|$ is odd, Alice is forced to colour w at some point, which, by the end of the game, cannot be part of a single connected red component due to Bob having coloured u or v in the first round. Thus, G is not A-perfect.

9.4 Conclusion

In this chapter we focused our attention on A-perfect graphs. Having already established that the density of the graph on which either game is played, plays an important role on deciding the outcome, we looked into A-perfect regular graphs. The interesting result in this direction is that A-perfect 3-regular graphs have bounded order, which is not true for d-regular graphs, for $d \geq 4$. We then proceed by providing sufficient conditions for a graph to be A-perfect. Apart from helping us better understand A-perfect graphs, the strategies we proposed to prove said conditions are by themselves interesting, as they shed some light on the behavior of the two games we have introduced.

Indeed, as we have seen in some graphs, notably in Section 9.3, some optimal strategies for Alice ensure that the red subgraph is connected at all times. This is the second time we encounter arguments implying the importance of whether Alice plays in a connected way or not (recall our discussion in Section 8.4), further enhancing our interest towards the study of a connected variant of the Maker-Breaker game. Such a variant could be defined similarly to the Maker-Breaker game, with the difference that Alice is always (except on her first turn) constrained to colour a neighbour of another red vertex, and the game ends when she cannot. Consequently, we could define $c_g^c(G)$ as the maximum score Alice can achieve in G when obliged to play in such a connected way. Clearly, $c_g^c(G) \leq c_g(G)$. We were able to observe that it is far from true that these two parameters are equal in general, even for some quite simple graphs. As an illustration, this is true for king's grids with only two rows and m columns (denoted by $P_2 \boxtimes P_m$). Indeed, in the connected case and for a sufficiently large m, consider the strategy according to which Bob colours the 4 vertices at distance 4 from the first vertex coloured by Alice. Bob following such a strategy guarantees that $c_g^c(P_2 \boxtimes P_m) = \mathcal{O}(1)$. On the other hand, in the non-connected case, consider the strategy where each time Bob colours a vertex v, Alice colours the neighbour of v in the other row. Alice following such a strategy guarantees that $c_g(P_2 \boxtimes P_m) = m$, which can be arbitrarily larger than the established $c_g^c(P_2 \boxtimes P_m)$.

Finally, it is worth mentioning that if Alice plays in a connected way in a Cartesian or king's grid, then the game becomes quite similar to the Angel and Devil Problem of Conway [35]. Optimal strategies for the devil in that game [35] allow to prove that $c_g^c(P_n \square P_m)$ is bounded above by an absolute constant.

Conclusion and perspectives

Chapter 10

Conclusion

In this thesis we have studied two families of combinatorial problems defined on graphs. In the first part of the thesis we dealt with proper labellings of graphs, and in the second we introduced and studied two variations of the largest connected subgraph game. The approach we followed in both parts is common. We first exhibited that the under consideration problems are computationally hard. Then we proceeded by providing either efficient algorithms to compute the corresponding parameters on restricted families of graphs, or bounds on the corresponding parameters for more general families of graphs. In my opinion, the importance of the work presented here stems from the introduction of the problems considered in the thesis.

In the first part of this thesis, we introduced and studied three new problems related to proper labellings of graphs, and provided answers to some open questions concerning the problem of finding equitable proper labellings. Notably, we were able to determine the complexity of finding an equitable proper labelling. Then, inspired by recent works that consider proper labellings that verify some additional constraints, we introduced the problem of finding k-labellings that apart from minimising k, also minimise the sum of the labels or the number of edges labelled 3. The three problems described in this paragraph form a nice group of problems, in the sense that better understanding the behaviour of one problem among them could lead to a better understanding of the other two. This is exhibited by the three new conjectures (Conjectures 2.1.2, 2.1.4 and 2.1.9) which we are led to propose, as well as the nice interplay that rises between them.

Let us comment on the problem of finding proper 3-labellings that also minimise the number of edges labelled 3. When dealing with that problem, we came up with various constructions of graphs that admit proper 3-labellings that assign label 3 on at least/most a fraction of their edges. Looking closely at the provided constructions, we can observe that the edges labelled 3 in the labellings we provide, sometimes define a matching of the graph. That is there are no two edges labelled 3 that are incident to a common vertex. Actually, we are no aware of any graph G with $\chi_\Sigma(G) = 3$ that does not verify this property. So, we ask ourselves whether this is a general phenomenon. If this turns out to be the case, then it could be interesting to study the minimum distance that separates the edges labelled 3 by any proper 3-labelling of the given graph. More generally, what can be said about the structures that are formed by the edges labelled 3 by any proper 3-labelling of a given graph?

The first part of this thesis closes with the introduction and study of the problem of finding a largest locally irregular induced subgraph. The behaviour of this problem shares some similarities with the problems considering proper labellings with additional constraints. Notably, we were able to provide an FPT algorithm parameterised by the treewidth and the maximum degree of the given graph. This was the second such algorithm we presented in this thesis, the first one being for the problem of finding proper k-labellings that minimise the sum of assigned labels. Actually, this FPT algorithm considering the treewidth and the maximum degree as the parameter, seems to be a recurring one for proper labellings. Indeed, such an algorithm was also exhibited in [31], where the authors consider the problem of finding proper labellings that also minimise the maximum induced colour. We stress that these algorithms are, at their core, quite similar. Thus, we ask ourselves whether it would be possible to formally define a general (Courcelle-like) framework that would guarantee and generate such algorithms.

The other aspect that is interesting about the parameterised complexity of the problem of finding a largest locally irregular induced subgraph of a given graph, is our proof that indicates that this problem is unlikely to be in FPT when parameterised only by the treewidth. The reduction in the proof of this result gives us a convenient way to control the degrees of the constructed graph. This is a good indication that it could actually be possible to prove a similar result for the problem of finding proper labellings that also minimise the sum of labels. We are once more led to ask whether this is also a result that can be generalised to cover more versions of proper labellings.

Of course, the big question about proper labellings is if the 1-2-3 Conjecture is true. It would even be interesting if we managed to prove this conjecture for more families of graphs, for example planar graphs. A first step towards such a result could be to show the 1-2-3 Conjecture for the family of planar graphs that admit a unique proper 4-colouring. This is a well characterised family [49] which is defined recursively: the smallest such graph is K_4. Then, starting with a planar embedding of a graph G_n of order n belonging in this family, to construct another member G_{n+1} of this family of order $n + 1$ we add one vertex v inside a face F of G_n and the three edges between v and the three vertices defining the face F (notice that by construction, each face of G_n is a triangle). This definition would be very useful in order to prove the 1-2-3 Conjecture for these graphs by, for example, doing induction on the order of the graph, or constructing a dynamic programming algorithm.

The second part of this thesis introduced and studied two versions of the largest connected subgraph game. Both versions of this game are natural problems that were not posed before. Moreover, the introduction of the Maker-Breaker version of this game follows naturally from the study of its scoring version.

Let us discuss further about an interesting aspect of both games, which has to do with connected strategies. Recall that in the Maker-Breaker game we came upon instances of graphs for which some optimal strategies for Alice required from her to colour at least one vertex which was not adjacent to an already red-coloured vertex, apart from the first vertex she coloured. We have no clear understanding of when Alice should favour such a strategy in general. We wonder, for instance, whether there is some structural characterisation of graphs for which any optimal strategy for Alice is not connected.

The situation becomes even more intriguing when considering the Scoring game. Indeed, in this version, it also makes sense to wonder about whether or not Bob's optimal strategies are connected as well. Actually, in the Scoring game we do not even have an equivalent of Lemma 8.3.1. That is, given a graph with many connected components, we are unsure of what Alice should do if Bob starts colouring vertices in an arbitrary connected component of the graph, and *vice versa*. In fact, this was a non-trivial obstacle that we had to overcome in order to arrive at the proof of Theorem 8.2.3, and we were only able to do so since cographs have a very particular and "exploitable" structure.

Finally, we would like to pose an additional direction of further research, apart from the open questions we left in Chapters 7, 8 and 9. Recall that the inspiration for introducing the Scoring game came from an attempt to model a real-life card game (explained in more details in Section 1.3). As a reminder, the scenario to be modeled is a game in which two players alternatively pick cards from a collection of common cards, with the goal of forming the most "synergistic" deck. As we already stated in the introduction of this thesis, the Scoring game we introduced is not a sufficient model to study the above scenario. Indeed, there is nothing to suggest that two pairs of synergistic cards that also share a card, form a triplet of synergistic cards. In other words, there is no guarantee that the player whose deck contains the cards corresponding to the largest connected subgraph of the graph defined by the shared collection of cards (the library graph), will indeed have a deck that is more synergistic than their opponent's. Moreover, we could easily imagine that some pairs of cards are more synergistic than others. This could be translated in the library graph by giving a weight on each edge, which would be a metric of how synergistic two cards are. So it would be very interesting to define a game similar to the largest connected subgraph game, but that is played on a weighted graph, and this would serve as a better model to study the above described drafting procedure (though the modeling would still be lacking).

Allow me to close this thesis on a more personal note. In the upcoming years I see my research continuing in the field of proper labellings, but also expanding in the general fields of combinatorial and algorithmic graph theory. In particular the questions of generalising our algorithmic results explained above seem very interesting to me. I have also enjoyed the few steps I have taken in the field of combinatorial games. Apart from the individual interest that each such game has, I find them appealing also for the value they can have in a didactic setting: most of these games are exceptionally easy to describe, fun to play with, but far from being trivially solved. Thus such games can be utilised as an ideal first introduction to combinatorial arguments, even for young students.

Bibliography

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

[2] L. Addario-Berry, K. Dalal, and B. A. Reed. Degree constrained subgraphs. *Electronic Notes in Discrete Mathematics*, 19:257–263, 2005.

[3] A. Agra, G. Dahl, T. A. Haufmann, and S. J. Pinheiro. The k-regular induced subgraph problem. *Discrete Applied Mathematics*, 222:14–30, 2017.

[4] Y. Alavi, A. Boals, G. Chartrand, O. Oellermann, and P. Erdős. k-path irregular graphs. *Congressus Numerantium*, 65, 01 1988.

[5] Y. Alavi, G. Chartrand, F. R. K. Chung, P. Erdős, R. L. Graham, and O. R. Oellermann. Highly irregular graphs. *Journal of Graph Theory*, 11(2):235–249, 1987.

[6] A. Ali, G. Chartrand, and P. Zhang. *Irregularity in Graphs*. Springer briefs in mathematics. Springer, 2021.

[7] S. D. Andres, M. Huggan, F. Mc Inerney, and R. J. Nowakowski. The orthogonal colouring game. *Theoretical Computer Science*, 795:312–325, 2019.

[8] K. Appel and W. Haken. Every planar map is four colorable. part I. Discharging. *Illinois Journal of Mathematics*, 21:429–490, 1977.

[9] K. Appel and W. Haken. Every planar map is four colorable. part II. Reducibility. *Illinois Journal of Mathematics*, 21:491–567, 1977.

[10] S. Arnborg and A. Proskurowski. Linear time algorithms for NP-hard problems restricted to partial k-trees. *Discrete Applied Mathematics*, 23(1):11–24, 1989.

[11] Y. Asahiro, H. Eto, T. Ito, and E. Miyano. Complexity of finding maximum regular induced subgraphs with prescribed degree. *Theoretical Computer Science*, 550:21–35, 2014.

[12] L. Babel. Triangulating graphs with few P_4's. *Discrete Applied Mathematics*, 89(1-3):45–57, 1998.

[13] L. Babel and S. Olariu. On the isomorphism of graphs with few P_4's. In *Proceedings of the 21st International Workshop on Graph-Theoretic Concepts in Computer Science (WG 1995)*, volume 1017 of *Lecture Notes in Computer Science*, pages 24–36. Springer, 1995.

[14] L. Babel and S. Olariu. On the p-connectedness of graphs - A survey. *Discrete Applied Mathematics*, 95(1-3):11–33, 1999.

[15] T. Bartnicki, J. Grytczuk, and S. Niwczyk. Weight choosability of graphs. *Journal of Graph Theory*, 60(3):242–256, 2009.

[16] O. Baudon, J. Bensmail, H. Hocquard, M. Senhaji, and É. Sopena. Edge weights and vertex colours: Minimizing sum count. *Discrete Applied Mathematics*, 270:13–24, 2019.

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

[18] O. Baudon, M. Pilśniak, J. Przybyło, M. Senhaji, É. Sopena, and M. Woźniak. Equitable neighbour-sum-distinguishing edge and total colourings. *Discrete Applied Mathematics*, 222:40–53, 2017.

[19] R. Belmonte and I. Sau. On the complexity of finding large odd induced subgraphs and odd colorings. *Algorithmica*, 83(8):2351–2373, 2021.

[20] J. Bensmail. *Partitions and decompositions of graphs*. Ph.d. thesis, Université de Bordeaux, 2014.

[21] J. Bensmail. *A contribution to distinguishing labellings of graphs*. Habilitation à diriger des recherches, Université côte d'azur, Dec. 2020.

[22] J. Bensmail, F. Fioravantes, and F. M. Inerney. On the role of 3's for the 1-2-3 conjecture. *Theoretical Computer Science*, 892:238–257, 2021.

[23] J. Bensmail, F. Fioravantes, and F. M. Inerney. On the role of 3's for the 1-2-3 conjecture. In T. Calamoneri and F. Corò, editors, *Algorithms and Complexity - 12th International Conference, CIAC, 2021, May 10-12, 2021, Proceedings*, volume 12701 of *Lecture Notes in Computer Science*, pages 103–115. Springer, 2021.

[24] J. Bensmail, F. Fioravantes, F. M. Inerney, and N. Nisse. The largest connected subgraph game. In L. Kowalik, M. Pilipczuk, and P. Rzazewski, editors, *Graph-Theoretic Concepts in Computer Science - 47th International Workshop, WG 2021, Warsaw, Poland, June 23-25, 2021, Revised Selected Papers*, volume 12911 of *Lecture Notes in Computer Science*, pages 296–307. Springer, 2021.

[25] J. Bensmail, F. Fioravantes, F. M. Inerney, and N. Nisse. The Largest Connected Subgraph Game. *Algorithmica (to appear)*, 2022.

[26] J. Bensmail, F. Fioravantes, F. Mc Inerney, and N. Nisse. Further results on an equitable 1-2-3 conjecture. *Discrete Applied Mathematics*, 297:1–20, 2021.

[27] J. Bensmail, F. Fioravantes, F. Mc Mc Inerney, N. Nisse, and N. Oijid. The Maker-Breaker Largest Connected Subgraph Game. Research report, Université Côte d'Azur, CNRS, Inria, I3S, Biot, France, 2021.

[28] J. Bensmail, F. Fioravantes, and N. Nisse. On proper labellings of graphs with minimum label sum. In L. Gasieniec, R. Klasing, and T. Radzik, editors, *Combinatorial Algorithms - 31st International Workshop, IWOCA 2020, Bordeaux, France, June 8-10, 2020, Proceedings*, volume 12126 of *Lecture Notes in Computer Science*, pages 56–68. Springer, 2020.

[29] J. Bensmail, F. Fioravantes, and N. Nisse. On proper labellings of graphs with minimum label sum. *Algorithmica*, 84(4):1030–1063, 2022.

[30] J. Bensmail, H. Hocquard, D. Lajou, and É. Sopena. A proof of the multiplicative 1-2-3 conjecture. In N. Balachandran and R. Inkulu, editors, *Algorithms and Discrete Applied Mathematics - 8th International Conference, CALDAM 2022, Puducherry, India, February 10-12, 2022, Proceedings*, volume 13179 of *Lecture Notes in Computer Science*, pages 3–14. Springer, 2022.

[31] J. Bensmail, B. Li, B. Li, and N. Nisse. On minimizing the maximum color for the 1-2-3 Conjecture. *Discrete Applied Mathematics*, 289:32–51, 2021.

[32] J. Bensmail, F. Mc Inerney, and K. Lyngsie Szabo. On $\{a, b\}$-edge-weightings of bipartite graphs with odd a, b. *Discussiones Mathematicae Graph Theory*, 42(1):159–185, 2022.

[33] J. Bensmail, M. Merker, and C. Thomassen. Decomposing graphs into a constant number of locally irregular subgraphs. *European Journal of Combinatorics*, 60:124–134, 2017.

[34] J. Bensmail, M. Senhaji, and K. S. Lyngsie. On a combination of the 1-2-3 conjecture and the antimagic labelling conjecture. *Discrete Mathematics and Theoretical Computer Science*, 19(1), 2017.

[35] E. R. Berlekamp, J. H. Conway, and R. K. Guy. *Winning Ways for Your Mathematical Plays, Volume 3*. A K Peters/CRC Press, 2018.

[36] P. Berman, M. Karpinski, and A. D. Scott. Approximation hardness of short symmetric instances of MAX-3SAT. *Electronic Colloquium on Computational Complexity*, (049), 2003.

[37] H. L. Bodlaender. On the complexity of some coloring games. *International Journal of Foundations of Computer Science*, 2(2):133–147, 1991.

[38] H. L. Bodlaender. A partial k-arboretum of graphs with bounded treewidth. *Theoretical Computer Science*, 209(1):1–45, 1998.

[39] E. Bonnet, F. Jamain, and A. Saffidine. On the complexity of connection games. *Theoretical Computer Science*, 644:2–28, 2016.

[40] R. L. Brooks. On colouring the nodes of a network. *Mathematical Proceedings of the Cambridge Philosophical Society*, 37(2):194–197, 1941.

[41] C. Browne. *Hex Strategy: Making the Right Connections*. AK Peters, 2000.

[42] C. Browne. *Connection Games: Variations on a Theme*. AK Peters, 2005.

[43] J. Bruno and L. Weinberg. A constructive graph-theoretic solution of the shannon switching game. *IEEE Transactions on Circuit Theory*, 17(1):74–81, 1970.

[44] L. Cai. Fixed-parameter tractability of graph modification problems for hereditary properties. *Information Processing Letters*, 58(4):171–176, 1996.

[45] L. Cao. Total weight choosability of graphs: Towards the 1-2-3-conjecture. *Journal of Combinatorial Theory, Series B*, 149:109–146, 2021.

[46] G. Chang, C. Lu, J. Wu, and Q. Yu. Vertex-coloring edge-weightings of graphs. *Taiwanese Journal of Mathematics*, 15:1807–1813, 2011.

[47] M. Changat, D. S. Lekha, I. Peterin, A. R. Subhamathi, and S. Spacapan. The median game. *Discrete Optimization*, 17:80–88, 2015.

[48] G. Chartrand, P. Erdős, and O. Oellermann. How to define an irregular graph. *The College Mathematics Journal*, 19:36–42, 1998.

[49] G. Chartrand and D. P. Geller. On uniquely colorable planar graphs. *Journal of Combinatorial Theory*, 6(3):271–278, 1969.

[50] G. Chartrand, M. Jacobson, J. Lehel, O. Oellermann, S. Ruiz, and F. Saba. Irregular networks. *Congressus Numerantium*, 64:197–210, 1986.

[51] J. Chen, B. Chor, M. Fellows, X. Huang, D. W. Juedes, I. A. Kanj, and G. Xia. Tight lower bounds for certain parameterized NP-hard problems. *Information and Computation*, 201(2):216–231, 2005.

[52] J. Chen, X. Huang, I. A. Kanj, and G. Xia. On the computational hardness based on linear FPT-reductions. *Journal of Combinatorial Optimization*, 11(2):231–247, 2006.

[53] J. Chen, X. Huang, I. A. Kanj, and G. Xia. Strong computational lower bounds via parameterized complexity. *Journal of Computer and System Sciences*, 72(8):1346–1367, 2006.

[54] M. Chlebík and J. Chlebíková. Complexity of approximating bounded variants of optimization problems. *Theoretical Compututer Science*, 354(3):320–338, 2006.

[55] V. Chvátal and P. Erdős. Biased positional games. *Annals of Discrete Mathematics 2*, pages 221–229, 1978.

[56] D. G. Corneil, Y. Perl, and L. K. Stewart. A linear recognition algorithm for cographs. *SIAM Journal on Computing*, 14(4):926–934, 1985.

[57] M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized Algorithms*. Springer, 2015.

[58] A. Dehghan, M.-R. Sadeghi, and A. Ahadi. Algorithmic complexity of proper labeling problems. *Theoretical Computer Science*, 495:25–36, 2013.

[59] R. Diestel. *Graph Theory, 4th Edition*, volume 173 of *Graduate texts in mathematics*. Springer, 2012.

[60] L. Ding, G.-H. Duh, G. Wang, T.-L. Wong, J. Wu, X. Yu, and X. Zhu. Graphs are $(1, \delta+1)$-choosable. *Discrete Mathematics*, 342(1):279–284, 2019.

[61] R. G. Downey and M. R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.

[62] É. Duchêne, V. Gledel, A. Parreau, and G. Renault. Maker-breaker domination game. *Discrete Mathematics*, 343(9), 2020.

[63] É. Duchêne, S. Gonzalez, A. Parreau, E. Rémila, and P. Solal. INFLUENCE: a partizan scoring game on graphs. *CoRR*, abs/2005.12818, 2020.

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

[65] P. Erdős and J. L. Selfridge. On a combinatorial game. *Journal of Combinatorial Theory, Series A*, 14(3):298–301, 1973.

[66] P. Erdős and R. J. Wilson. On the chromatic index of almost all graphs. *Journal of Combinatorial Theory, Series B*, 23(2-3):255–257, 1977.

[67] S. Even and R. E. Tarjan. A combinatorial problem which is complete in polynomial space. *Journal of the ACM*, 23(4):710–719, 1976.

[68] R. Faudree and J. Lehel. Bound on the irregularity strength of regular graphs. *Combinatorics, Colloquium Mathematical Society János Bolyai*, 52:247–256, 1987.

[69] F. Fioravantes, N. Melissinos, and T. Triommatis. Complexity of finding maximum locally irregular induced subgraphs. In A. Czumaj and Q. Xin, editors, *18th Scandinavian Symposium and Workshops on Algorithm Theory, SWAT, 2022, June 27-29, 2022, Tórshavn, Faroe Islands*, volume 227 of *LIPIcs*, pages 24:1–24:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022.

[70] A. M. Frieze, R. J. Gould, M. Karoński, and F. Pfender. On graph irregularity strength. *Journal of Graph Theory*, 41(2):120–137, 2002.

[71] D. Gale. The game of Hex and the Brouwer fixed-point theorem. *American Mathematical Monthly*, 86:818–827, 1979.

[72] M. Gardner. *The Scientific American Book of Mathematical Puzzles & Diversions*. Simon and Schuster, N.Y., 1959.

[73] M. Gardner. *The Second Scientific American Book of Mathematical Puzzles and Diversions*. Simon and Schuster, N.Y., 1961.

[74] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.

[75] H. Grötzsch. Zur Theorie der diskreten Gebilde. *VII. Ein Dreifarbensatz für dreikreisfreie Netze auf der Kugel. Wiss. Z. Martin-Luther-Univ. Halle-Wittenb, Math-Natwiss Reihe*, 8:109–120, 1958.

[76] P. J. Heawood. Map-colour theorems. *Quarterly Journal of Mathematics, Oxford*, 24:332–338, 1890.

[77] D. Hefetz, M. Krivelevich, M. Stojaković, and T. Szabó. *Positional Games*, volume 44 of *Oberwolfach Seminars*. Springer, Basel, 2014.

[78] M. Kalkowski. *Metody algorytmiczne w badaniach siły nieregularności grafów*. Ph.d. thesis, Uniwersytet im. Adama Mickiewicza, Wydzial Matematyki i Informatyki, 2010.

[79] M. Kalkowski, M. Karoński, and F. Pfender. Vertex-coloring edge-weightings: Towards the 1-2-3-conjecture. *Journal of Combinatorial Theory, Series B*, 100(3):347–349, 2010.

[80] M. Kalkowski, M. Karoński, and F. Pfender. A new upper bound for the irregularity strength of graphs. *SIAM Journal on Discrete Mathematics*, 25(3):1319–1321, 2011.

[81] M. Karoński, T. Łuczak, and A. Thomason. Edge weights and vertex colours. *Journal of Combinatorial Theory, Series B*, 91(1):151–157, 2004.

[82] R. Keusch. Vertex-coloring graphs with 4-edge-weightings. *CoRR*, abs/2202.02788, 2022.

[83] S. Khot and V. Raman. Parameterized complexity of finding subgraphs with hereditary properties. *Theoretical Computer Science*, 289(2):997–1008, 2002.

[84] H. A. Kierstead and W. T. Trotter. Planar graph coloring with an uncooperative partner. *Journal of Graph Theory*, 18(6):569–584, 1994.

[85] R. Küsters. *Memoryless determinacy of parity games*, volume 2500 of *Lecture Notes in Computer Science*. Springer, 2002.

[86] J. Köbler, U. Schöning, and J. Torán. *The Graph Isomorphism Problem*. Progress in Theoretical Computer Science. Birkhäuser Boston, MA, 1993.

[87] D. König. Über graphen und ihre anwendung auf determinantentheorie und mengenlehre. *Mathematische Annalen*, 77(4):453–465, 1916.

[88] U. Larsson, R. J. Nowakowski, J. P. Neto, and C. P. Santos. Guaranteed scoring games. *Electronic Journal of Combinatorics*, 23(3), 2016.

[89] U. Larsson, R. J. Nowakowski, and C. P. Santos. Game comparison through play. *Theoretical Compututer Science*, 725:52–63, 2018.

[90] U. Larsson, R. J. Nowakowski, and C. P. Santos. Games with guaranteed scores and waiting moves. *International Journal of Game Theory*, 47(2):653–671, 2018.

[91] J. M. Lewis and M. Yannakakis. The node-deletion problem for hereditary properties is NP-complete. *Journal of Computer and System Sciences*, 20(2):219–230, 1980.

[92] C. N. Lintzmayer, G. O. Mota, and M. Sambinelli. Decomposing split graphs into locally irregular graphs. *Discrete Applied Mathematics*, 292:33–44, 2021.

[93] D. Lokshtanov, D. Marx, and S. Saurabh. Lower bounds based on the exponential time hypothesis. *Bulletin of the European Association for Theoretical Computer Science*, 105:41–72, 2011.

[94] H. Lu, Q. Yu, and C.-Q. Zhang. Vertex-coloring 2-edge-weighting of graphs. *European Journal of Combinatorics*, 32(1):21–27, 2011.

[95] K. S. Lyngsie. On neighbour sum-distinguishing $\{0,1\}$-edge-weightings of bipartite graphs. *Discrete Mathematics and Theoretical Computer Science*, 20(1), 2018.

[96] P. Micek and B. Walczak. A graph-grabbing game. *Combinatorics, Probability and Computing*, 20(4):623–629, 2011.

[97] B. Mohar. Face covers and the genus problem for apex graphs. *Journal of Combinatorial Theory, Series B*, 82(1):102–117, 2001.

[98] H. Moser and D. M. Thilikos. Parameterized complexity of finding regular induced subgraphs. *Journal of Discrete Algorithms*, 7(2):181–190, 2009.

[99] P. Mukkamala and D. Pálvölgyi. Drawing cubic graphs with the four basic slopes. In M. van Kreveld and B. Speckmann, editors, *Graph Drawing*, pages 254–265, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[100] W. Mulzer and G. Rote. Minimum-weight triangulation is NP-hard. *Journal of the ACM*, 55, 2006.

[101] J. Nešetřil, A. Raspaud, and É. Sopena. Colorings and girth of oriented planar graphs. *Discrete Mathematics*, 165-166:519–530, 1997.

[102] M. Polanyi. *Personal Knowledge: Towards a Post-Critical Philosophy*. University of Chicago Press, 1958.

[103] J. Przybyło. Irregularity strength of regular graphs. *Electronic Journal of Combinatorics*, 15, 06 2008.

[104] J. Przybyło. On decomposing graphs of large minimum degree into locally irregular subgraphs. *Electronic Journal of Combinatorics*, 23(2):2–31, 2016.

[105] J. Przybyło. The 1-2-3 conjecture almost holds for regular graphs. *Journal of Combinatorial Theory, Series B*, 147:183–200, 2021.

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

[107] J. Przybyło and M. Woźniak. Total weight choosability of graphs. *Electronic Journal of Combinatorics*, 18(1), 2011.

[108] M. L. Rahman and T. Watson. 6-uniform maker-breaker game is PSPACE-complete. In *Proceedings of the 38th International Symposium on Theoretical Aspects of Computer Science (STACS 2021)*, pages 57:1–57:15, 2021.

[109] S. Reisch. Hex ist PSPACE-vollständig. *Acta Informatica*, 15(2):167–191, 1981.

[110] N. Robertson, D. P. Sanders, P. D. Seymour, and R. Thomas. The four-colour theorem. *Journal of Combinatorial Theory, Series B*, 70(1):2–44, 1997.

[111] T. J. Schaefer. On the complexity of some two-person perfect-information games. *Journal of Computer and System Sciences*, 16(2):185–225, 1978.

[112] B. Seamone. The 1-2-3 Conjecture and related problems: a survey. *CoRR*, abs/1211.5122.

[113] M. Senhaji. *Neighbour-distinguishing decompositions of graphs*. Ph.d. thesis, Université de Bordeaux, Sept. 2018.

[114] A. Shapovalov. Occupation games on graphs in which the second player takes almost all vertices. *Discrete Applied Mathematics*, 159(15):1526–1527, 2011.

[115] J. Skowronek-Kaziów. Multiplicative vertex-colouring weightings of graphs. *Information Processing Letters*, 112(5):191–194, 2012.

[116] C. Thomassen, Y. Wu, and 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.

[117] V. G. Vizing. On an estimate of the chromatic class of a p-graph. *Diskretnyi Analiz*, 3:25–30, 1964.

[118] B. Vučković. Multi-set neighbor distinguishing 3-edge coloring. *Discrete Mathematics*, 341(3):820–824, 2018.

[119] W. Wang, Y. Bu, M. Montassier, and A. Raspaud. On backbone coloring of graphs. *Journal of Combinatorial Optimization*, 23(1):79–93, 2012.

[120] W. Wang and K. Lih. List Coloring Halin Graphs. *Ars Combinatoria*, 77(10):53–63, 2005.

[121] T. Wong and X. Zhu. Total weight choosability of graphs. *Journal of Graph Theory*, 66(3):198–212, 2011.

[122] J. Wu and X. Zhu. Lower bounds for the game colouring number of partial k-trees and planar graphs. *Discrete Mathematics*, 308(12):2637–2642, 2008.

[123] X. Zhu. Refined activation strategy for the marking game. *Journal of Combinatorial Theory, Series B*, 98(1):1–18, 2008.

[124] X. Zhu. Every nice graph is $(1, 5)$-choosable. *CoRR*, abs/2104.05410, 2021.

Étiquetages d'arêtes, colorations de sommets et jeux combinatoires sur des graphes

Foivos-Sotirios FIORAVANTES

Résumé

Cette thèse considère deux familles de problèmes définis sur des graphes : les étiquetages d'arêtes propres et les jeux combinatoires. Nous traitons ces problèmes de façon similaire (et classique) : nous montrons que les problèmes considérés sont difficiles à résoudre, puis nous trouvons des algorithmes efficaces sur des instances restreintes.
Nous nous concentrons d'abord sur des problèmes concernant des étiquetages propres de graphes. Pour un entier k fixé, un *k-étiquetage* d'un graphe G est une fonction associant à chaque arête de G une étiquette parmi $\{1, \dots, k\}$. Un k-étiquetage induit une coloration des sommets de G, où chaque sommet reçoit comme couleur la somme des étiquettes de ses arêtes incidentes. Un k-étiquetage est *propre* si, dans la coloration induite, deux sommets adjacents de G reçoivent des couleurs différentes. D'après la Conjecture 1-2-3, tout graphe connexe d'ordre au moins 3 admet un 3-étiquetage propre. Nous considérons trois variantes de cette conjecture. Nous étudions les k-étiquetages propres équilibrés, pour lesquels les étiquettes assignées apparaissent dans les mêmes proportions. La deuxième variante concerne les étiquetages propres qui minimisent la somme des étiquettes utilisées. Enfin, nous nous intéressons aux 3-étiquetages propres qui minimisent le nombre de fois où l'étiquette 3 est attribuée. Le choix d'étudier ces variantes est naturel. En effet, une version équilibrée de la Conjecture 1-2-3 est que presque tous les graphes G admettent un 3-étiquetage propre équilibré. En outre, la somme des étiquettes d'un tel étiquetage est au plus égale à $2|E(G)|$ et associe l'étiquette 3 à au plus un tiers des arêtes de G. Nous prouvons que les problèmes d'optimisation introduits sont NP-difficiles. Grâce à des résultats structurels et algorithmiques, nous sommes amenés à proposer de nouvelles conjectures pour ces problèmes, que nous vérifions sur quelques classes de graphes (complets, bipartis, réguliers, 3-chromatiques, etc.). Notre travail renforce l'idée que des variantes plus fortes de la Conjecture 1-2-3 pourraient être vraies. Nous terminons en considérant le problème consistant à trouver un plus grand sous-graphe induit d'un graphe donné qui admet un 1-étiquetage propre. Il est prouvé que ce problème est difficile à résoudre et qu'il n'est pas approximable à un facteur $\mathcal{O}(|V(G)|^{1-\frac{1}{c}})$ près pour tout entier c. Néanmoins, nous fournissons des algorithmes paramétrés efficaces.
La deuxième partie de la thèse introduit le *jeu du plus grand sous-graphe connexe Maker-Breaker*, joué par deux joueurs, Alice et Bob, sur un graphe G, initialement non coloré. Les joueurs colorent à tour de rôle les sommets de G, chacun avec sa couleur, jusqu'à ce que tous les sommets soient colorés. Alice est la gagnante si, à la fin, le plus grand sous-graphe connexe de G induit par sa couleur est d'ordre au moins k, un entier fixé. Sinon, Bob gagne le jeu. Nous considérons aussi une version Score du même jeu, dans laquelle le gagnant est le joueur dont la couleur induit le plus grand sous-graphe connexe de G à la fin du jeu. Nous prouvons que décider de l'issue de ces deux jeux est PSPACE-difficile et nous fournissons des algorithmes efficaces pour le cas où le jeu se déroule dans certaines familles de graphes (chemins, cycles, cographes, $(q, q-4)$-graphes, etc.). En comparant ces deux jeux, la principale différence que nous observons est que Bob ne peut jamais gagner la version Score (si Alice joue de manière optimale). Pour une valeur de k égale à la moitié de l'ordre de G, remarquons que si Alice peut gagner la version Maker-Breaker alors elle peut aussi construire un sous-graphe connexe du même ordre dans la version Score ; de tels graphes sont nommés A-parfaits. Nous étudions les graphes réguliers qui sont A-parfaits et prouvons que tout graphe 3-régulier A-parfait a au plus 16 sommets. Nous terminons en fournissant des conditions suffisantes pour qu'un graphe soit A-parfait.

Mots-clés : Graphe, Coloration, Conjecture 1-2-3, Jeux combinatoires, Complexité.

Abstract

In this thesis, we consider two families of computational problems defined on graphs: proper edge-labellings and combinatorial games. We attack these problems in a similar (and classical) way: we show that they are computationally hard, and then find efficient algorithms for instances with specific structure.
First we focus on problems related to proper labellings of graphs. For some natural number k, a *k-labelling* is a weight function on the edges of a graph G, assigning weights, called labels in this context, from $\{1, \dots, k\}$. A k-labelling induces a vertex-colouring of G, where each vertex receives as colour the sum of the labels of its incident edges. A k-labelling is *proper* if the induced vertex-colouring is proper, *i.e.*, such that any two adjacent vertices of G are assigned different colours. According to the so-called 1-2-3 Conjecture, any connected graph of order at least 3 should admit a proper 3-labelling. We consider three variations of this conjecture. We look into equitable proper k-labellings, for which the assigned labels appear an equal number of times. We then focus on proper labellings that also minimise the sum of labels being used, and finally, proper 3-labellings that also minimise the number of times that the label 3 is assigned. The choice to study these variations is natural. Indeed, an equitable version of the 1-2-3 Conjecture claims that almost every graph G should admit an equitable proper 3-labelling. Also, the sum of labels of such a labelling would be at most $2|E(G)|$ and it would assign label 3 to at most one third of the edges of G. We prove that the introduced optimisation problems are NP-hard. Furthermore, through structural and algorithmical results, we propose new conjectures for the upper bounds of the parameters that we study, which we verify for specific graph classes (*e.g.* complete, bipartite, regular, 3-chromatic, etc.). Interestingly, our work gives further evidence that stronger variations of the 1-2-3 Conjecture could hold. We close our study of proper labellings by considering the problem of finding a largest induced subgraph of a given graph that admits a proper 1-labelling. This problem is proven to be computationally hard and not approximable within a ratio of $\mathcal{O}(|V(G)|^{1-\frac{1}{c}})$ for every natural number c. Nevertheless, we provide efficient parameterised algorithms.
In the second part of the thesis, we introduce and study the *Maker-Breaker largest connected subgraph game*. This game is played by two players, Alice and Bob, on a shared, initially uncoloured graph G. The two players take turns colouring the vertices of G, each one with their own colour, until there remains no uncoloured vertex. Alice is the winner of the game if, by the end, the largest connected subgraph of G induced by her colour is of order greater than k, where the natural number k is also given at the start of the game. Otherwise Bob wins the game. We also consider a Scoring version of the same game, played in the same way, but in which the winner is the player whose colour induces the largest connected subgraph of G by the end of the game. We first prove that deciding the outcome of both of these games is PSPACE-hard, and then proceed by providing efficient algorithms when the games are played on particular graph classes (*e.g.* paths, cycles, cographs, $(q, q-4)$-graphs, etc.). Comparing the behaviour of these games, one of the main differences we observe is that Bob can never win the Scoring version (if Alice plays optimally). Nevertheless, if Alice can win the Maker-Breaker version when playing on G for a value of k equal to half the order of G (the best outcome she can hope for), then she can build a connected subgraph of the same order for the Scoring version; such graphs are called A-perfect. We then study regular graphs that are A-perfect and prove that any 3-regular A-perfect graph has order at most 16. We finish by providing sufficient conditions for a graph to be A-perfect.

Keywords: Graph, Colouring, 1-2-3 Conjecture, Combinatorial games, Complexity.