THÈSE POUR OBTENIR LE GRADE DE DOCTEUR DE L'UNIVERSITÉ DE MONTPELLIER

En Informatique

École doctorale : Information, Structures, Systèmes

Unité de recherche : LIRMM, UMR5506

Forced structures and connectivity problems in digraphs

Présentée par Amadeus REINALD
le 12 Septembre 2025

Devant le jury composé de

Stéphane BESSY Professeur, HdR, LIRMM, Université de Montpellier	**Directeur de thèse**
Daniel GONÇALVES Chargé de recherche, HdR, LIRMM, Université de Montpellier, CNRS	**Directeur de thèse**
Alexandre PINLOU Professeur, HdR, LIRMM, Université de Montpellier	**Président du jury**
Alantha NEWMAN Chargée de recherche, HdR, G-SCOP, Université Grenoble-Alpes, CNRS	**Rapportrice**
Jean-Sébastien SERENI Directeur de recherche, HdR, ICUBE, Université de Strasbourg, CNRS	**Rapporteur**
Marthe BONAMY Chargée de recherche, HdR, LABRI, Université de Bordeaux, CNRS	**Examinatrice**
Pierre CHARBIT Maître de conférences, HdR, IRIF, Université Paris-Cité	**Invité**

UNIVERSITÉ DE MONTPELLIER

Forced structures and connectivity problems in digraphs

Amadeus Reinald

Acknowledgements

Je me dois d'ouvrir ces lignes en remerciant mes encadrants, Daniel et Stéphane, envers qui ma gratitude est immense. Merci pour votre investissement sans relâche du début à la fin, pour l'optimisme constant que vous reflétiez, et pour la liberté que vous m'avez laissée de toucher à tout. Merci Stéphane, pour ta pédagogie qui a su (enfin) me faire aimer les graphes dirigés, pour tes connaissances inépuisables, et pour ton intuition qui oriente encore aujourd'hui la mienne. Merci Daniel, de t'être autant consacré à ces questions étrang(èr)es, d'avoir toujours partagé tes pistes fraîches, et de m'avoir permis de prendre autant de recul à travers ta vision. Je réalise un peu plus chaque jour la chance que j'ai eue d'avoir en vous des directeurs aussi dévoués, de toujours trouver vos portes ouvertes, de pouvoir puiser de vos intuitions complémentaires, et de ressortir (ré)inspiré de chacune de nos discussions. Je vous suis infiniment reconnaissant de m'avoir permis de m'épanouir sous votre conseil, que je continuerai d'honorer au mieux.

Je tiens ensuite à remercier Alantha Newman et Jean-Sébastien Sereni d'avoir accepté de rapporter ma thèse avec autant d'attention. Vos retours, jusqu'aux parties les plus reculées du texte, ont grandement contribué à faire émerger ce manuscrit dans la forme que j'espérais. Merci à Alexandre Pinlou d'avoir assumé la présidence de mon jury, ainsi qu'à Marthe Bonamy et (de fait) Pierre Charbit d'avoir accepté d'être mes examinateurs. C'est un honneur pour moi que chacun d'entre vous ait porté autant d'intérêt à mon travail, et d'avoir pu défendre devant vous.

Durant ces trois années, j'ai eu la chance de faire partie de l'équipe AlGCo, où sur chaque tableau les traits de craie pouvaient susciter ma curiosité. Je tiens d'abord à remercier les permanents, qui m'ont accueilli dans leurs couloirs, et m'ont permis de les considérer miens : Christophe, Emeric, Ignasi, Marin, Mathieu, Matthieu, Mickael, Pascal, Petru (mulțumesc pentru sfaturi), et William (merci pour le moulin, et le café brûlé). J'ai aussi été entouré de nombreux camarades de route, devenus amis, qui animaient mes journées (et soirées) au labo, au Charlie's et à la grimpe. Merci à Hugo, d'être toujours (un peu trop) partant pour gribouiller sur notre tableau (ou en terrasse) ; à Gaétan pour ta joie contagieuse, parfois insouciante (mais de quand même m'avoir sauvé du Cévenol) ; à Laure pour tous les gâteaux d'anniversaire (malgré des désaccords toujours très constructifs) ; à Yann pour ta générosité (même si je n'accroche toujours pas à la chanson française) ; à Simon de m'avoir fait replonger dans l'ingénierie aéronautique ; à Giannos de nous avoir fait découvrir tous les spots de Montpellier ; and to Vagelis for our discussions about everything from life to work. Merci également à ceux que j'ai croisés plus brièvement, mais qui ont largement contribué à faire vivre l'atmosphère de l'équipe : Ariane, Guilherme, Florent, Raul, Sebastian et Thomas. Je suis aussi redevable envers tous les professeurs de l'Université de Montpellier et de l'IUT Montpellier-Sète qui m'ont fait confiance pour intervenir dans leurs matières, me permettant d'y découvrir mon épanouissement dans l'enseignement. Enfin, je n'aurais pas pu m'adonner pleinement à mon travail sans l'aide de nos assistants. Merci Laetitia d'avoir systématiquement pris le temps pour le moindre de mes soucis avec autant de sympathie. Nicolas, ta disparition a laissé un si grand vide pour nous tous. Merci pour ta gentillesse désintéressée, et pour ton dévouement aux autres, que j'ai senti aussi bien quand je n'étais qu'un énième stagiaire de passage, que lors de ma toute dernière mission de thèse.

Mon parcours aurait sûrement pris une tournure tout autre sans certains professeurs qui m'ont encouragé à des moments pivotaux, je remercie donc à souhait Mme Guiton, M. Lacarra, Eric Thierry et Pascal Koiran. Je suis reconnaissant envers Stéphan, qui a su me guider avec autant d'enthousiasme durant mon mémoire de master, alors que j'y faisais mes premiers pas en graphes. Merci ensuite à Dimitrios d'avoir encouragé mon acharnement même quand le labo s'apprêtait à fermer. Merci Frédéric de m'avoir permis de faire quelques pas de plus à Sophia, bien que je n'aie pas touché au barbecue. Thank you Sang-Il, for welcoming me so warmly in Daejeon and bringing me to Jeju.

Heureusement, la communauté des graphes est un monde très collaboratif, où l'espoir comme le désespoir sont souvent à partager. Pour commencer, je suis très reconnaissant envers celles et ceux m'ayant permis de voyager à travers le monde sans jamais être (trop) dépaysé. Merci à Amer et Nicolas B. de m'avoir permis de travailler à Beyrouth, Marthe à Bordeaux (même si je ne sais pas faire lever un gâteau), EunJung and Kazuhiro in Nagoya, Seokbeom and Sang-Il in Daejeon, Ugo et Bartosz à Cracovie, ainsi que Lucas et Ken-Ichi à Tokyo. Ensuite, je tiens à remercier mes nombreux (autres) co-auteurs, qui m'ont dévoilé de leurs perspectives autant de facettes de notre domaine : Caroline S., Carmen, Clément R., Édouard, Filip, Florian, Hector, Hoang, Jonathan, Jørgen, Lech, Marcin, Nicolas N., Noleen, Quentin C., Rémi, Ross, Théo, Thomas, Tomáš et Uéverton. J'ai aussi énormément appris des discussions avec mes collaborateurs et mes amis à travers toute notre communauté, entre autres Aline, Amir, Arthur, Ben, Carl, Clément D., Clément L., Cléophée, Colin, Edgar, Eileen, Foivos, Francesco, Grégoire, Guillaume A., Igor, Julien, Kevin, Lucas D.M., Maël, Mathis, Nikolaos, Pierre A., Quentin D., Rutger, Sarah, Samuel, Sena, Timothé, Valentin et Zhihan.

Cette aventure aurait été bien morose sans mes amis de tous bords, qui ont ponctué mon quotidien de diversions bienvenues. Merci à Lucas de toujours accepter d'attaquer nos conjectures de 18h autour d'un café (trop cher) ou d'un soup curry (trop épicé). Merci à Ugo de toujours être chaud pour parcourir les montagnes et les bars, malgré mon emploi du temps de ministre (sic). Thank you Linda, for your support when I arrived in Korea, and for your selfless guidance ever since. Thank you Seokbeom for your eagerness to collaborate and hang out in Daejeon, Jungho for welcoming me to Seoul, Donggyu for the hikes around Korea, and to all of you for the makgeolli and the LoL nights at the PCBang. Merci à Kenza pour les (dé)tours à travers l'Ecusson, to Aurora for all the hikes and museums, to Tuan-Anh for bringing me to Vietnam, et à Maximos pour les karaokés et autres badinages. Merci à mes autres amis de l'ENS, à Charles, Jérémy, Nemo, Victor, et Yoann, pour des raclettes toujours plus dramatiques, ainsi qu'à Alain, Edwige, Hadrien, Louis-Sébastien, Maxime, Sophian et Vincent. Merci à Swann pour tous ces bons moments à Lyon, et les égarements à la campagne, ainsi qu'à Guillaume et Antoine pour toutes les soirées autour d'un St-Nectaire.

Je suis infiniment redevable à ma famille, sans qui je ne serais arrivé nulle part. Merci à vous tous pour votre soutien si précieux, des rentrées scolaires et autres ennuis, jusqu'au jour de ma soutenance. Merci à Nicoleta, à qui je dois tout, à Iris, de croire en moi avec autant d'insistance, et à Angel, d'illuminer chacune de mes journées. Merci encore à Iris pour les beaux dessins préfaçant les Chapitres 3, 5, et 8, et à Angel pour les jolis dessins des Chapitres 4 et 7. Je n'aurai franchement pas pu rêver d'illustrations plus charmantes (et cohérentes), que je suis honoré de pouvoir inclure ici (en plus ça gardera les lecteurs).

Je finirai par des remerciements envers les organismes qui m'ont soutenu financièrement, me donnant ainsi la liberté de travailler sans attaches, sur ce qui me passionne vraiment. Pour ceci, merci à l'ENS de Lyon de m'avoir accordé un contrat doctoral, à l'ANR Digraphes d'avoir financé mes visites et trois de mes stages pré-doctoraux, ainsi qu'à l'équipe AlGCo, l'école doctorale I2S et le LIRMM.

Abstract

This thesis explores the structure of directed graphs in relation to colouring, as well as algorithmic problems regarding their connectivity.

Over the last century, a central topic of graph theory has been the study of substructures forced by large chromatic number. These range from Hadwiger's conjecture from 1943, to the Gyárfás-Sumner conjecture (1975), and at the extreme end Brooks' theorem. The first part of this thesis deals with directed variants of these questions. In 1980, Burr conjectured that a bound of $2k-2$ on the chromatic number would ensure all oriented trees of order k as subdigraphs, and showed a quadratic bound. We begin by proving the first subquadratic bound for Burr's conjecture. Then, we settle a special case of a Gyárfás-Sumner analogue proposed by Aboulker, Charbit and Naserasr in 2020. Namely, large dichromatic number forces all induced orientations of the 4-vertex path in sparse digraphs. Our last contribution to these questions is an analogue of Brooks' theorem for a more general notion of (di)colourings.

Computational questions looking to increase or to decrease the connectivity of digraphs have a long history in computer science. In the second part of this thesis, we consider variants of the well studied NP-hard problems of strong connectivity augmentation and feedback arc set. We first look at making a plane oriented graph strongly connected by adding a minimal number of arcs, while preserving planarity. We show that this problem remains NP-hard, but admits a fixed-parameter tractable (FPT) algorithm. Our second problem, generalizing feedback arc set, aims at rendering a digraph acyclic by inverting subsets of arcs. There, we give upper bounds on the number of inversions required for digraphs of a given order, as well as FPT algorithms and hardness results for tournaments.

Résumé

Cette thèse explore la structure des graphes dirigés relativement à leur colorations, ainsi que des problèmes algorithmiques concernant leur connectivité.

Au cours du dernier siècle, un aspect central de la théorie des graphes a été l'étude de sous-structures forcées par un nombre chromatique élevé. Ces questions vont de la conjecture de Hadwiger de 1943, à celle de Gyárfás-Sumner (1975), et à l'extrême, le théorème de Brooks. La première partie de cette thèse s'intéresse aux variantes orientées de ces questions. En 1980, Burr a conjecturé qu'une borne de 2k sur le nombre chromatique assurerait la présence de tous les arbres orientés de taille k comme sous-digraphes, et a montré une borne quadratique. Nous commençons par prouver la première borne sous-quadratique pour la conjecture de Burr. Ensuite, nous établissons un cas particulier d'un analogue de la conjecture de Gyárfás-Sumner proposé par Aboulker, Charbit et Naserasr en 2020. Plus précisément, les digraphes épars de nombre dichromatique suffisamment élevé contiennent nécessairement toute orientation induite du chemin à 4 sommets. Notre dernière contribution sur ces questions est la preuve d'un analogue du théorème de Brooks pour une notion plus générale de (di)coloration.

En informatique théorique, les problèmes cherchant à augmenter ou à réduire la connectivité d'un digraphe ont une longue histoire. Dans la seconde partie de cette thèse, nous considérons des variantes de deux problèmes bien étudiés, l'augmentation fortement connexe, et feedback arc set. On s'intéresse d'abord au problème cherchant à rendre un graphe orienté fortement connexe à travers l'ajout d'un nombre d'arcs minimal, tout en préservant la planarité. Notre deuxième problème, généralisant feedback arc set, vise à rendre un digraphe acyclique en renversant des ensembles d'arcs. Nous prouvons des bornes supérieures sur le nombre d'inversions requises pour les digraphes d'un ordre donné, ainsi que des algorithmes FPT et des preuves de difficulté algorithmique.

Contents

Chapter 1

Introduction

This chapter is a general introduction to graphs and directed graphs, oriented towards the main structural and algorithmic concerns of this thesis. On the one hand, we motivate the structural study of graphs with high chromatic number, further investigated in the introductory Chapter 2 towards Part I. On the other hand, we highlight fundamental algorithmic problems on (di)graph connectivity, and continue this thread in the introduction Chapter 6 to Part II.

1.1 An introduction to Graph Theory

1.1.1 Preliminaries

Mathematical notation

For any set X, we let $|X|$ denote the size of X. Given another set Y, we use $X \setminus Y$ and $X - Y$ interchangeably, as well as $X \cup Y$ and $X + Y$. We will abuse this notation to hold also for elements $x \in X$, that is $X - x = X \backslash \{x\}$, and $X + x = X \cup \{x\}$. The set $\binom{X}{k}$ consists of all subsets of X of size k, while 2^X represents all subsets of X. We let $A \uplus B$ denote the disjoint union of A and B. If R is an equivalence relation for X, and $x \in X$, $[x]_R$ refers to the equivalence class of x. Then, the quotient of X by R, denoted X/R, is the set of equivalence classes of R in X. For an integer n, we let $[n]$ be the set of integers that are at most n. For two integers $i < j$, the interval $[i, j]$ consists of all integers greater than or equal to i and lower than or equal to j. The notation log refers to the base-2 logarithm $\log_2$. Given two positive-valued functions f, g on the integers, we write $f = O(g)$ when there exists an n_0 and a constant C, such that for any $n \geq n_0$, we have $f(n) \leq Cg(n)$. We write $f = o(g)$ to mean $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$.

Graphs

A graph is defined by vertices and edges, which can be seen as abstractions of objects and their relations, respectively. Formally, a graph $G = (V, E)$ is given by a set of *vertices* V and a set of unordered vertex pairs E, called *edges*. When G is given, we will typically use V and E to refer to its vertices and edges, and sometimes $V(G)$ and $E(G)$ to avoid ambiguity. All graphs considered in this thesis are finite, and, unless stated otherwise, *simple*, meaning they forbid loops $\{x, x\}$, and two ("parallel") copies of the same edge. The *order* of a graph is defined as $n(G) = |V|$, its *size* is given by $m(G) = |E|$. We will often use $n = n(G)$ and $m = m(G)$.

Two vertices x, y that are joined by an edge $e = \{x, y\} \in E$ are *adjacent*. Then, x and y are the endpoints of e, x is a neighbour of y and vice-versa. We will alternatively write xy or yx to mean $\{x, y\}$ when there is no ambiguity. A *walk* between $x_1 \in V$ and $x_k \in V$ is a sequence of edges $x_1x_2, x_2x_3, \ldots, x_{k-1}x_k$. When no edges repeat, this is called a *trail*, and when no vertices repeat, it is a *path*. When $x_1 = x_k$, these definitions correspond to a *closed walk*, a *circuit* and a *cycle* respectively.

The (open) *neighbourhood* $N(v)$ of a vertex v is the set of its neighbours. Its *closed neighbourhood* is defined as $N[v] = N(v) \cup \{v\}$. We extend these definitions to subsets of vertices $X \subseteq V$, where $N(X) = \bigcup_{v\in X} N(v) \setminus X$, and $N[X] = \bigcup_{v\in X} N[v]$. A *graph class* is a set of graphs, and often refers to the set of graphs satisfying a given property.

Isomorphism

As graphs are meant to be abstractions of the relations between objects, most of the time, only edges matter, and not which specific objects they are joining. To formalize this, we want to identify a graph with all graphs that have the "same" edges. Formally, two graphs G, H are *isomorphic* if there exists a bijective mapping $f : V(G) \to V(H)$, the *isomorphism*, such that $\{u, v\} \in E(G)$ if and only if $\{f(u), f(v)\} \in E(H)$. We represent this by writing $G \simeq H$, and note that $\simeq$ defines an equivalence relation on all graphs. All chapters of this thesis, except for Chapter 8, consider graphs up to isomorphism, and G will in fact always refer to $[G]_\simeq$, its equivalence class. We

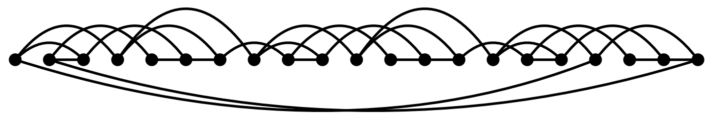


Figure 1.1: A graph embedded as an arc diagram.

typically consider vertices as distinct elements $V = \{v_1, \ldots, v_n\}$ without specifying their nature, or let $V = [1, n]$.

Algorithms and complexity

As a central data structure in computer science, graphs are the objects of many algorithmic problems. We measure the *complexity* of an algorithm by the (maximal) number $f(n)$ of elementary operations used, across all instances of size n. In most problems, given a solution, we can at least *verify* that it is correct in polynomial-time. These make up the class NP (for Nondeterministic Polynomial time). Among those, the class P consists of problems for which a solution can be *found* in polynomial-time, which are considered efficient regardless of the polynomial. For example, the problem of computing the shortest path between two vertices is in P. There are problems, called *NP-complete*, which have been shown to "simulate" any problem in NP, through polynomial-time *reductions*. For example, determining the existence of a Hamiltonian path, visiting every vertex of a graph exactly once, is a famous NP-complete problem. The widely admitted P≠NP conjecture, states that not all problems in NP are in P, and would certify our belief that NP-complete problems do not admit polynomial-time algorithms.

1.1.2 Graph drawings and planarity

Graphs can be graphed. This is in fact the first way they were introduced in Euler's 1741 paper on the bridges of Königsberg [Eul41]. To this day, this visual aspect drives a large portion of modern research. A *drawing* of a graph represents vertices by points and edges as line segments, or even curves, joining pairs of them. To emphasize the absence (or the deletion) of an edge, we may draw it in dashed lines. Formally, a drawing is an *embedding* of a graph on a surface, which for our purposes will always be the plane. The same graph can be embedded in many ways, but we insist each are a representation of the same graph: only the adjacency relation matters. We may for instance lay out all vertices on a common line, then add the corresponding edges between them as curves above it. This kind of representation is called an arc diagram, see Figure 1.1 for an example. The drawing of the graph above contains multiple *edge crossings*, that is, edges intersecting elsewhere than at their endpoints (vertices). This complicates our reading, but the same graph turns out to be embeddable without any crossing, as shown in Figure 1.2. Such a drawing, avoiding crossing edges, is called a *plane embedding*. For a plane (embedded) graph, the different regions delimited by areas of the plane between edges are called *faces*. To manipulate a face, we associate it with the closed subwalk following its border. Graphs admitting a plane embedding are *planar*, but this is hardly the case of all graphs (see Figure 1.5). Planar graphs are in fact a very restricted class, on which many questions become simpler.

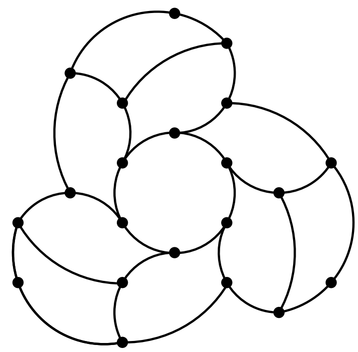


Figure 1.2: The same graph with a plane (non-crossing) embedding.

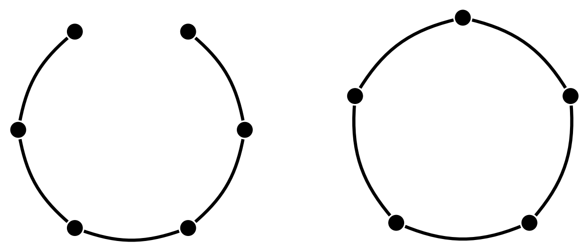


Figure 1.3: The 6-path and the 5-cycle

1.1.3 Basic graphs

Paths, cycles, and connectivity

We will now show a few simple graphs, as examples, but also because of their role in many results and conjectures on general graphs. In the following, we will set $V = \{v_1, \ldots, v_n\}$ and specify the edges corresponding to each graph over V.

Starting with natural definitions, the *path* of order n, or n-path, is defined as $P_n = (V, E)$ with $E = \{v_i v_{i+1} : 1 \leq i < n\}$, see Figure 1.3.

Given a graph G, two vertices $u, v \in V$ are *connected* if they are joined by a path. Otherwise, they are disconnected. Then, G is *connected* if any pair of its vertices is connected. Maximal subsets of pairwise connected vertices of G form (induce) its *connected components*, which partition $V(G)$. We will have a deeper look at connectivity in Subsection 1.1.7.

The *cycle* of order n, or n-cycle, adds one edge to the path to reconnect its endpoints. That is, we define it as $C_n = (V, E)$ with $E = \{v_i, v_{i+1[n]} : 1 \leq i \leq n\}$, see Figure 1.3. The *length* of a path or a cycle is its size. Using cycles, we can express a stronger notion of connectivity, where two vertices are 2-*connected* if they are part of a common cycle, and the whole graph is when all vertex pairs are.

Trees and Forests

A graph without cycle is a called a *forest*, and connected forests are called *trees*, see Figure 1.4. Trees (and forests) are simple structurally, and for instance, the absence of cycles enables dynamic

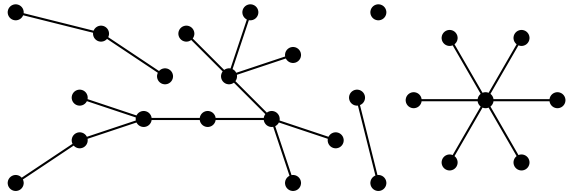


Figure 1.4: This whole graph is a forest, and each connected component is a tree.

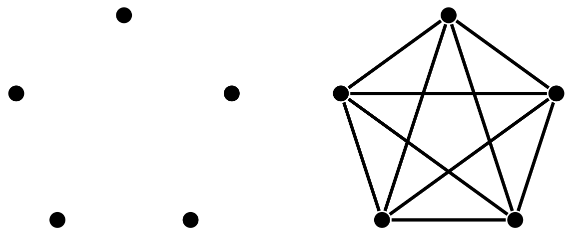


Figure 1.5: The anticomplete graph I_5, and the clique K_5, the smallest non-planar graph.

programming algorithms for a wide range of problems. For this reason, they have been extended to "tree-like" graphs through the notion of *treewidth*, a measure of the proximity of a graph to a tree (see [Cyg+15]).

Complete and anticomplete graphs

Definition-wise, the two simplest graphs are those containing either all or none of all possible edges. A *complete* graph, or *clique*, on V is defined as $K_n = (V, \binom{V}{2})$, containing $\binom{n}{2}$ edges. An *anticomplete*, or *independent* graph on V is defined as $I_n = (V, \emptyset)$. Sitting in between these two graphs are *bicliques*, or *complete bipartite graphs*, denoted $K_{n,p}$. These have vertex set $V = A \uplus B$, with $|A| = n$, $|B| = p$, no edge lies fully within A or B, and all np edges between A and B are present.

1.1.4 Operations

In both practical and theoretical settings, it is often of interest to study operations modifying a graph at hand. We are usually asked to transform a graph into another, or to modify it such that it satisfies some wanted property. Let us start with the simplest operations, on a single vertex or edge. Given a graph $G = (V, E)$, the deletion of a vertex v yields the graph $G - v = (V - v, E - \{e \in E : v \in e\})$. Conversely, one may add a vertex to v and choose the adjacencies towards the rest of the graph. The *deletion* of an edge e yields the graph $G - e = (V, E - e)$. Conversely, the (edge) *augmentation* of G by some $e \in \binom{V}{2} \setminus E$ yields $G + e = (V, E + e)$.

We can also define operations involving pairs of vertices. Given two vertices $x, y \in V$, the *identification* of x and y replaces them with a vertex z adjacent to all their neighbours. That is, identifying x and y yields the graph $G/\{x, y\}$ with vertex set $V - \{x, y\} + z$ and edge set $E - \{e \in E : x \in e \vee y \in E\} + \{zw : w \in N(x) \cup N(y)\}$. We extend this terminology to subsets $X \subseteq V$, where identifying X yields the graph obtained by (any) sequence of pairwise identifications of vertices in X, ending in a single vertex. More generally, given an equivalence relation R on V, the quotient graph G/R consists in vertex set V/R along with edges between pairs of classes admitting adjacent members. In our uses, R will most often come from a partition of V. The *contraction* of an edge $e = xy$ deletes e, and identifies x and y. We extend the definition of contraction to a connected subset of vertices X to mean its identification, since we may contract edges of X ending in a single vertex (see also Figure 1.6, where blue clouds represent contractions). The *subdivision* of an edge $e = xy$ consists in removing e, and adding a new vertex w in-between, with edges xw and wy. A graph H is a subdivision of G if it can be obtained from G by a sequence of subdivisions. An *immersion* of a graph H within G is the given of an injective mapping f from $V(H)$ to $V(G)$, along with internally edge-disjoint paths in G between $f(x)$ and $f(y)$ for every $xy \in V(H)$.

There are also more "global" operations, acting on the whole graph, or even combining two graphs into a new one. The *complement* of a graph G is defined as $\overline{G} = (V, \binom{V}{2} \setminus E)$. Given two graphs G, H, possibly sharing some vertices, we may want to combine these two into a new graph. The simplest way to do so is by taking their *union*, defined as $G \cup H = (V(G) \cup V(H), E(G) \cup E(H))$. If G and H have disjoint vertex sets, their *join* is defined by taking their union, and adding all edges between $V(G)$ and $V(H)$. Identifying a vertex $u \in V(G)$ and a vertex $v \in V(H)$ yields the graph $G \cup H/\{u, v\}$, which can be seen as "gluing" G at vertex u to H at v. The *substitution* of a vertex $v \in V(G)$ with a graph H consists in replacing v with $V(H)$, along with all edges of H, and for any vertex in $V(H)$, adding edges towards all neighbours of v. When H is anticomplete, this operation is called a *blow-up*.

1.1.5 Relations

To understand the structure of a graph, it is natural to want to compare it to other graphs. We have seen the equivalence relation of isomorphism, but we will be more interested in containment relations. These allow us to give a precise meaning to the vague notion of *sub*structures. Equivalently, we may consider graphs forbidding some substructure (understood as those forbidding it for a specific relation). Relations can also be seen as (partial) orderings among graphs, which also raises interesting questions. We consider a graph $G = (V, E)$ to present the three main relations on graphs: induced subgraphs, subgraphs, and minors, from most restrictive to least. The first two will be our main consideration in Part I.

Considering any $X \subseteq V$, the subgraph of G *induced* by X, is defined as $G[X] = (X, E \cap \binom{X}{2})$. A subgraph which can be obtained this way is called an *induced subgraph* of G, see the first graph of Figure 1.6. Equivalently, induced subgraphs are those obtainable from G by (successive) vertex deletions. Then, a graph class is said to be *hereditary* if it is closed under taking induced subgraphs.

A *subgraph* H of G is a graph such that $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$, see the second graph in Figure 1.6. In that case, we write $H \subseteq G$. Equivalently, subgraphs are those obtainable from G by (successive) vertex deletions and edge deletions. Remark that induced subgraphs are also subgraphs. When referring specifically to a subgraph that is a tree or a path, we will use the terms *subtree* and *subpath*. Then, a graph class is said to be *monotone* if it is closed under taking subgraphs.

A *minor* of a graph G is any graph that may be obtained from G by vertex or edge deletions, as well as edge contractions, see the third graph in Figure 1.6. Note that this is the most permissive notion, as (induced) subgraphs are also minors. A class of graphs which is closed under taking minors is simply called *minor-closed*.

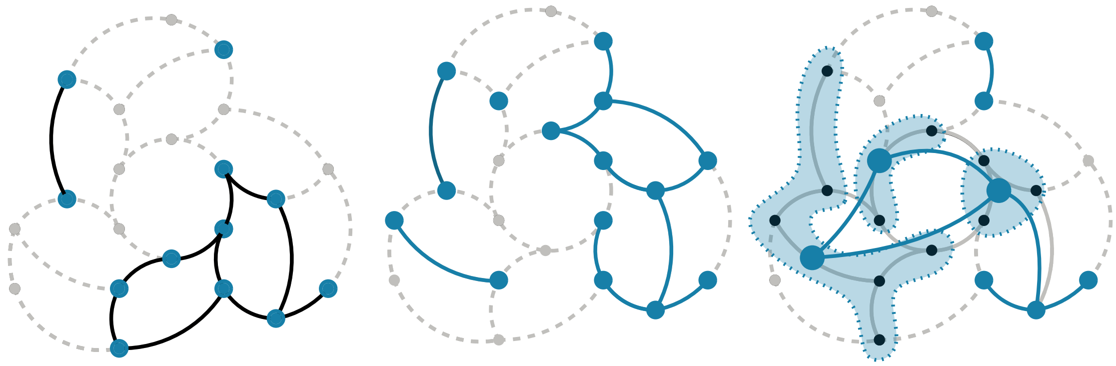


Figure 1.6: An induced subgraph, a subgraph, and a minor of the graph in Figure 1.2, re-drawn below in (dashed) grey. Highlighted in blue are the objects preserved in defining substructures: vertices for the induced subgraph; vertices and edges for the subgraph; vertices, edges and contracted connected sets of vertices (shown by the blue dotted clouds) for minors.

1.1.6 Properties and parameters

In the following, we define basic qualitative and quantitative properties, or invariants, to describe a graph G. Typically, quantitative properties are referred to as *parameters* of the graph. Many questions look at comparing parameters by bounding one in terms of (a function) of the other. We will often describe the value of parameters for induced subgraphs. There, for a parameter p and a subset $X \subseteq V(G)$, we abuse notation by using $p(X) = p(G[X])$.

Degrees

The *degree* of a vertex $v \in V(G)$ is the number of its neighbours, denoted $d(v) = |N(v)|$. The *handshaking lemma*, which Euler already noticed in [Eul41], relates the sum of degrees with the number of edges. Indeed, any $e = xy$ contributes exactly twice in the following:

$$\sum_{v \in V(G)} d(v) = 2m$$

The *maximum degree* of G is defined as $\Delta(G) = \max_{v \in V(G)} d(v)$, its *minimum degree* is defined as $\delta(G) = \min_{v \in V(G)} d(v)$. Its *average degree* is then $\mathrm{ad}(G) = (\sum_{v \in V} d(v))/n = \frac{2m(G)}{n(G)}$, thanks to the handshaking lemma. Then, the *maximum average degree* is $\mathrm{mad}(G) = \max_{H \subseteq G} ad(H)$. The *degeneracy* of G is the minimum integer k such that every subgraph of G has a vertex of degree at most k. Then, a k-*degeneracy ordering* is an order on $V(G)$ such that every vertex is adjacent to at most k predecessors. It follows immediately that, if $\mathrm{mad}(G) \leq k$, then G is k-degenerate. Conversely, if G is d-degenerate, any subgraph $H \subseteq G$ satisfies $m(H) \leq dn(H)$, so $\mathrm{mad}(G) \leq 2d$.

A less trivial observation which is often useful is that high average degree implies a minimum degree (induced) subgraph.

Observation 1.1.1. *If a graph G satisfies $ad(G) \geq 2k$, then it contains a subgraph with minimum degree k.*

Indeed, consider any vertex v such that $d(v) < k$. Then, $m(G - v) > \frac{\text{ad}(G)n}{2} - k = k(n-1)$, meaning $\text{ad}(G - v) \geq 2k$. Therefore, the successive removal of vertices with degree less than k finishes with a graph of minimum degree at least k.

Structure

Often times, properties are characterized using substructures of G, either through their own form, or because of a certain interaction with the rest of the graph. The *distance* between two vertices $x, y \in V(G)$ is the length of a minimum subpath of G between x and y, and infinite if none exist. The *girth* of a graph is the length of the minimum cycle it contains, and is infinite if the graph is a forest. The *diameter* of a graph is the maximal distance between all pairs of vertices. A *spanning subtree* of G is a subtree containing all vertices, note that any minimal connected subgraph of G yields such a tree. Moving on to induced subgraphs, an *independent set*, or *stable set*, of G, is an independent induced subgraph. The order of the largest independent set is the *independence number* of G, denoted $\alpha(G)$. A *clique* of G is a complete (induced) subgraph, and the order of the largest clique is called the *clique number* of G, denoted $\omega(G)$.

A *dominating set* of a graph G is a subset $U \subseteq V$ such that any $w \in V \setminus U$ has a neighbour in U. The size of a smallest dominating set in G is its *domination number* denoted $\gamma(G)$. A vertex is *universal* (or an apex) if it is adjacent to all others, and *isolated* if it is adjacent to none.

1.1.7 Connectivity

Given a pair of vertices, asking whether they are connected is probably the most natural question we can come up with. What if we now want to define a quantity to express wider ranges of connectivity? There are two natural ways to express that a pair of vertices is "highly" connected. The first one is to say there are many ways to connect them, which can sensibly be interpreted as many (disjoint) paths. The second one is a negative definition, saying they are hard to disconnect, with few vertex or edge deletions. Strikingly, these two interpretations are dual in a very tight sense, as the associated values are equal, through the following theorems of Menger.

Vertex connectivity

The *local connectivity* between $x, y \in V(G)$, denoted $\kappa(x, y)$ is the maximum number of internally vertex-disjoint paths between them. The term *internally* expresses that those paths share only their endpoints. Then, x and y being connected corresponds to $\kappa(x, y) \geq 1$. Now, a *x, y-separator* is some $X \subseteq V$ such that x and y are disconnected in $G - X$.

Menger's theorem, which we state below, is the foundational result on graph connectivity, highlighting the duality between separators and vertex-disjoint paths [Men27].

Theorem 1.1.2 (Menger '27)**.** *Let G be a graph and $x, y \in V(G)$ two non-adjacent vertices. The size of a minimum separator between x and y equals $\kappa(x, y)$, the maximum number of internally vertex-disjoint paths between x and y.*

Then, G is *k-vertex-connected*, or simply *k-connected*, if for any $x, y \in V$, we have $\kappa(x, y) \geq k$. Note that being 1-connected corresponds exactly to being connected.

Edge connectivity

The *local edge connectivity* between x and y, denoted $\lambda(x, y)$, is the maximum number of edge-disjoint paths between them. Note that $\lambda(x, y) \geq \kappa(x, y)$. Then, a *cut* between x and y is a subset $F \subseteq E$ such that x and y are disconnected in $G - F$. Menger showed cuts and edge-disjoint paths enjoy the same duality as in the vertex case.

Theorem 1.1.3 (Menger '27)**.** *Let G be a graph and $x, y \in V(G)$, the size of a minimum cut between x and y equals $\lambda(x, y)$, the maximum number of edge-disjoint paths between x and y.*

Similarly, we then say G is *k-edge-connected* if for any $x, y \in V$, we have $\lambda(x, y) \geq k$. This notion also coincides with connectedness for $k = 1$, though there are 2-edge-connected graphs which are not 2-(vertex-)connected.

Disconnection problems

The edge version of Menger's theorem is a special case of the more general duality between minimum (weighted) cuts and maximal flows, as shown by Ford and Fulkerson [FF56]. As such, there is a polynomial-time algorithm to compute a minimum cut between two vertices, also due to the same authors. Their method starts with an arbitrary collection of edge-disjoint paths (a flow) and employs an auxiliary directed graph to find "augmenting paths", whose combination with the initial collection increases their number. The vertex-connectivity version also reduces to a max flow problem by splitting every vertex into an edge, yielding polynomial-time solvability as well.

Rather than disconnecting a single pair of vertices, what if we wish to disconnect G as a whole? If we take this to mean finding a minimum set of edges (or vertices) whose deletion disconnects G, or reduces its connectivity below a certain value, this is rather simple. We can iterate over all pairs of vertices and select the lowest minimum cut (or separator), which will also be minimum for the sake of disconnecting G.

An intrinsically different question arises when we impose that connectivity decreases *everywhere* in G, say below a given threshold k.

Question 1.1.4. *What is the minimum number of deletions resulting in a graph where no pair of vertices is k-(edge)-connected?*

If the goal is to prevent any two vertices from being connected at all, we are asking for the resulting graph to be anticomplete. For vertex deletions, this is the same as searching for a minimum set of vertices incident to all edges of the graph, which is the Vertex Cover problem. This is a fundamental NP-hard problem, which is well-studied and admits a polynomial-time algorithm for any fixed solution size, as well as a 2-approximation. For edge deletions, this is not very interesting as all edges would have to be removed, but the story is different for the directed analogue. In the introduction to Part II, we continue this thread with *feedback* problems, which correspond Question 1.1.4 for 2-(edge)-connectivity.

Connection problems

The converse question is also interesting, how do we increase the connectivity of G? Note that here, the question is already a global one by definition, and the connectivity between every pair should increase. If we allow vertex addition, we may add multiple universal vertices, and it is clear that both vertex and edge connectivity increase across all (initial) vertices of G. A more "well posed" problem is to only allow edge augmentations.

Question 1.1.5. *What is the minimum number of edge augmentations making a graph k-(edge)-connected?*

These *connectivity augmentation* questions have a long history, and we discuss them more deeply the introduction to Part II. There, we will also be interested in restrictions on the resulting graph, such as maximal degree, and especially for us, planarity.

1.1.8 Colourings

Another way to handle the structural complexity of a graph is to split it into simpler parts. The first question of this kind is one of Guthrie, from 1852, asking whether the regions of every (planar) map can be coloured with four colours such that no two neighbours are of the same colour. To see why this is really a graph question, associate each region to a vertex, and link two regions if they share a border. It is easy to see that the resulting graph is planar, and that the question is to split its vertices into four independent sets.

A (proper) *colouring* of a graph G is the assignment of a colour to each vertex such that no two adjacent vertices are of the same colour. That is, a colouring $G = (V, E)$ with k colours, or k-colouring, partitions V into subsets $V_1, \dots, V_k$ each inducing an independent set. Then, the *chromatic number* $\chi(G)$ of a graph G is the minimum integer k for which such a partition exists. We will say vertices in V_i are those "coloured" i, and V_i is a *colour class*, or even *colour* for short. A subset of vertices is *monochromatic* if all have the same colour. We note that these notions can also be adapted to define edge colourings, but this is not the goal here. A graph G is k-critical if it has chromatic number k, but any of its subgraphs has chromatic number at most $k-1$.

In turn, the conjecture of Guthrie is equivalent to asking whether every planar graph is 4-colourable. This stood for more than a century, and became probably the most famous problem in graph theory, before Appel and Haken gave a computer-assisted proof [AH76] (see [Rob+97] for a shorter proof).

Theorem 1.1.6 (Four Colour Theorem, Appel, Haken '76)**.** *Every planar graph is* 4*-colourable.*

Upper bounds

A simple way to obtain a proper colouring of a graph is to greedily colour vertices one by one. Say we colour with integers, and at each step, we colour the new vertex with the lowest integer that is not present among its neighbours. This immediately yields a $\Delta + 1$-colouring of our graph, but most of the time Δ colours turn out to be enough, according to Brooks' theorem [Bro41].

Theorem 1.1.7 (Brooks '41)**.** *If G is a connected graph, then $\chi(G) \le \Delta$, unless G is an odd cycle or a clique.*

This is already a fundamental result in graph colouring, but in the scope of this thesis, it is the first example of structures forced by (very) high chromatic number (here, relative to Δ). This greedy colouring approach can in fact be used whenever the degeneracy is bounded, which is why degeneracy is also called *colouring number*.

Observation 1.1.8. *For any k-degenerate graph G, $\chi(G) \le k+1$. The same applies if $mad(G) \le k$.*

Indeed, take a minimal counter-example G to the above, and let $v \in V(G)$ be such that $d(v) \le k$. By minimality, we can colour $G - v$ with colours in $[1, k+1]$, then colour v with an element of

$[1, k+1]$ that is not already taken by its neighbours, which contradicts $\chi(G) > k+1$. In fact, planar graphs are 5-degenerate, which by the above already yields a bound of $\chi \leq 6$ for the four-colour theorem.

Lower bounds

A trivial way to lower bound $\chi(G)$ is to find a subgraph of G with large chromatic number. The clique K_k is perhaps the simplest such object: every pair of vertices is adjacent, so $\chi(K_k) = k$. Then, we obtain the following.

Observation 1.1.9. *For any graph G, $\chi(G) \geq \omega(G)$.*

Observe that C_5 contains no triangle yet has chromatic number 3, so clearly there are cases when $\chi > \omega$.

Since we are partitioning G into independent sets, it is natural to ask for a dependency on the independence number α. And indeed, since each colour class is of size at most α, we get the following.

Observation 1.1.10. *For any graph G, $\chi(G) \geq \frac{n(G)}{\alpha(G)}$.*

Algorithms

In all generality, computing the chromatic number of a graph G is NP-hard, even if G is planar and we ask for 3-colorability, see Garey et al. [GJS76]. Of course, we may always colour G with $\Delta + 1$ colours in linear time through the greedy approach described above. It is also possible to produce a Δ-colouring in linear-time when G is not one of the obstructions to Brooks' theorem, as shown by Baetz and Wood [BW14], building on a proof of Lovász [Lov75]. Thus, considering graphs such that $\chi \geq \Delta$ is one of the few restrictions under which deciding chromatic number (between Δ and $\Delta + 1$) is polynomial.

1.1.9 Minor structure

Connectivity and chromatic number give ways to measure the simplicity of a graph, but what is it that makes a graph simple in general? We cannot satisfyingly ask this for a specific graph, since any answer would be relative, so we should rather ask what "simple" classes of graphs are, and justify this simplicity. In this subsection, we overview the well-known theory of minor structure, and its consequences. This is outside the scope of our results, but it is a fundamental development of structural graph theory, where minors seem to be the most well-behaved notion. Results for minors routinely raise questions about generalizations to other containment notions.

As a first non-trivial example of a simple class, we may consider the class of trees, which are simple thanks to the absence of cycles as subgraphs. A more elegant way to express this is that trees are exactly the graphs forbidding C_3 as a minor, and now trees are expressed through a single *obstruction*. A larger class which we still feel is simple is planar graphs, but for a different reason based in their geometry. Surprisingly, they too admit a definition of the same nature, as they are exactly the class of graphs forbidding K_5 and $K_{3,3}$ as minors, which is known as Kuratowski's theorem [Kur30]. The unification of these two seemingly distant simplicities through *forbidden minors* characterizations begs a generalization to other classes.

To go beyond planarity, we may consider graphs which can be drawn without crossings, but now allowing for other surfaces than the plane. Robertson and Seymour show that for any g, the class of graphs embeddable on a surface of genus at most g is characterized by a finite list of forbidden minors [RS90]. This list is not known for any surface other than the plane, and already for the torus ($g = 1$), it is in the excess of 17535 minors, see Myrvold and Woodcock [MW18].

For a less geometric example, consider *treewidth*, which measures the proximity of a graph to a tree. Let us start by formally defining it.

Definition 1.1.11. A *tree decomposition* of a graph G is the given of a tree T on vertices, or *bags*, which are subsets $X_i \subseteq V(G)$, satisfy the following:

1. $\bigcup_i X_i = V(G)$,
2. If $v \in X_i \cap X_j$, then v belongs to any X_ℓ on the path from X_i to X_j in T,
3. For every $(u, v) \in E(G)$, there exists a bag X_i containing (u, v).

Then, the width of a tree-decomposition T is defined as $\max_i |X_i| - 1$. The *treewidth* of G is the minimum width over all of its tree-decompositions.

Graphs of treewidth at most k are well understood in terms of minors, through their relation with grids. The $k \times k$ grid is the graph on $[1, k]^2$, where vertices (i, j) and (i', j') are joined whenever exactly one of their indices differs by 1. Robertson and Seymour [RS86] showed that the $k \times k$ grid has treewidth k, and most crucially, that the following loose converse holds.

Theorem 1.1.12 (Robertson, Seymour '86)**.** *There exists a function f such that every graph with treewidth at least $f(k)$ contains the $k \times k$ grid as a minor.*

This is the cornerstone to many theoretically efficient algorithms (actually, FPT, see Subsection 1.3.1) for problems where large grids certify a yes or no answer, as we may then assume all grids are small and solve the problem using a "tree decomposition".

All classes above are minor-closed, and in fact, any minor-closed class can equivalently be defined by the (possibly infinite) set of graphs which do not belong to it. So the remarkable part is the finiteness of the obstructions. The graph minor structure theorem of Robertson and Seymour [RS04] is probably the most important result in graph structure, as a far-reaching generalization of all results above.

Theorem 1.1.13 (Robertson, Seymour '04)**.** *Every minor closed class of graphs can be defined by a finite set of forbidden minors.*

This includes a large variety of classes defined structurally, and the proof of this result also uses a generalization of tree-decompositions for graphs forbidding any minor, witnessing that graphs forbidding a minor are indeed "simple". Theorem 1.1.13 also has deep algorithmic consequences. Indeed, consider any problem on a graph G asking for a solution of size at most k, and for which taking minors does not increase it. This means that positive instances form a minor-closed class. Then, our problem is equivalent to asking whether one of the forbidden *obstructions* is present as a minor in G. Such a test can be done in polynomial-time, as shown by Robertson and Seymour [RS95], and recently brought down to near linear-time by Korhonen, Pilipczuk and Stamoulis [KPS24]. Since there is only a finite list of forbidden minors (for a given problem and fixed k), this shows a polynomial-time algorithm exists. Still, there is a catch, which is that there is no constructive method to build the set of obstructions, so this is only an existence proof, and cannot produce an actual algorithm. Still, efficient algorithms have been devised for plenty of such problems, either by using Theorem 1.1.12, or through ad-hoc techniques.

1.1.10 Substructures and chromatic number

To get a better grasp on chromatic number, we could first look for alternative characterizations. In terms of other parameters, all our upper bounds on χ are (qualitatively) justified by bounded degeneracy, but there is no hope for even a loose converse because of bicliques. In the following, we look at chromatic number through the lens of structure, and first investigate whether a characterization by forbidden structures exists.

It is easy to see that the class of graphs satisfying $\chi \leq k$ is not minor-closed, so Theorem 1.1.13 is of no help to obtain a characterization by forbidden minors. Such a characterization is hopeless, even asymptotically, as 1-subdivisions of sufficiently large cliques contain any graph as a minor, yet have $\chi = 2$. Large cliques are a certificate of high chromatic number only when they appear as subgraphs. Then, can we characterize k-colourable graphs by forbidden (induced) subgraphs? This class is both monotone and hereditary, so we may indeed define it as graphs forbidding the infinite list of graphs with $\chi = k + 1$. The question now becomes: can this list be made finite? That is, are its (induced) minimal graphs finite? The answer is negative already for $k = 2$, where all odd cycles are incomparable obstructions, which generalizes to larger values as well. In both cases, a satisfactory characterization is impossible, but we can still ask which structures are necessary for high χ, even if they do not have high χ themselves.

Question 1.1.14. *What structures are forced to appear in graphs with high chromatic number?*

Taking the contrapositive, we are equivalently asking which forbidden structures bound χ.

Forcing minors

Setting the full characterization aside, we turn to asking which minors can be found in graphs of large chromatic number. Recalling Kuratowski's characterization of planar graphs as those forbidding K_5 and $K_{3,3}$ as a minor, we may wonder if forbidding them independently bounds χ. It turns out that forbidding K_5 already implies $\chi \leq 4$, as Wagner showed this to be a consequence of the four-colour theorem. This lead Hadwiger to conjecture the generalization of this result to K_k-minor free graphs.

Conjecture 1.1.15 (Hadwiger '43)**.** *Let G be a graph such that $\chi(G) \geq k$, then G contains K_k as a minor.*

This is the strongest statement we can ask for, as K_k shows that no graphs on more than k vertices may be obtained, and minors of K_k include any graph on at most k vertices. Despite considerable effort, this is only known to hold for $k \leq 6$ thanks to Robertson et al. [RST93]. The best bound for Hadwiger's conjecture is due to Delcourt and Postle, showing that if $\chi(G) \geq O(k \log\log(k))$, then G contains K_k as a minor [DP25].

Towards forcing subgraphs

The existence of K_k as a subgraph implies $\chi \geq k$, but odd cycles having $\chi = 3$ show us that we cannot always expect K_k to appear provided $\chi = k$. Morally, this does not yet imply cliques are not responsible for high χ. They may still be, in a weaker sense, if for any clique K_k, there exists a value of χ that ensures K_k appears.

This possibility was ruled out in the 1950s, through constructions showing high χ cannot even force a triangle (K_3) to appear as a subgraph. Perhaps the simplest counter-example is the one of Mycielski, dating 1955 [Myc55].

Theorem 1.1.16 (Mycielski '55)**.** *There exist K_3-free graphs of arbitrarily large chromatic number.*

The recursive construction of Mycielski produces a sequence of graphs, which, through a kind of blow-up operation at each step, manages to simulate the addition of a universal vertex, increasing χ without creating triangles. There are plenty of other constructions of triangle-free graphs with large chromatic number, such as that of Blanche-Descartes [Des54], Zykov [Zyk49], and Burling [Bur65]. A more recent and elegant construction was given by Bonnet et al. [Bon+23].

To overcome this obstacle, graph theorists have resorted to studying classes of graphs in which χ is indeed tied to ω. Here, we only consider hereditary classes, as otherwise any such class with arbitrarily large chromatic number would contain arbitrarily large cliques. This means the class would include all graphs, which we have seen cannot satisfy our requirement. In fact, Erdös and Hajnal conjectured that any monotone class of graphs with unbounded χ contains a subclass with arbitrarily large χ and girth [Erd88]. Therefore, we investigate hereditary classes where χ is tied to ω, first by asking for equality, then by allowing for some function f such that $\chi \leq f(\omega)$.

Forcing cliques

The strongest property we could ask of a class in this respect is for the chromatic number to be exactly the clique number. Since we are considering hereditary classes, this means we should impose $\chi = \omega$ for every induced subgraph. Graphs satisfying the above are called *perfect*, a term coined by Berge in 1961 [Ber61]. He observed that induced odd cycles, as well as induced complements of odd cycles, are always an obstacle to having $\chi = \omega$. This led to him conjecturing that these are in fact the only obstacles, meaning every graph without an induced odd cycle nor an induced complement of an odd cycle is perfect. This was only resolved in 2002 by Chudnovsky et al. [Chu+06] and is now known as the Strong perfect graph theorem.

What if we now allow some room between χ and ω? Considering the length of [Chu+06], it is probably hopeless to give an exact characterization of classes for which $\chi = \omega + i$ for fixed $i \geq 1$. Still, we can ask more qualitatively for classes $\mathcal{C}$ admitting a function f such that $\chi \leq f(\omega)$ in $\mathcal{C}$.

Question 1.1.17. *For which classes of graphs is the chromatic number bounded by a function of the clique number?*

Formally a class of graphs $\mathcal{C}$ is *χ-bounded* if there exists a function f, called the χ-binding function, such that for any $G \in \mathcal{C}$, $\chi(G) \leq f(\omega(G))$.

An important subclass of perfect graphs are interval graphs, obtained from a set of segments drawn on a line by considering those as vertices, and joining them whenever they intersect. Generalizing this, we may consider intersection graphs of segments in the *plane*. Erdös asked in the 1970s whether these segment intersection graphs are χ-bounded (see [Gyá87]), which was only disproven in 2014 by Pawlik et al. [Paw+14]. A natural χ-bounded subclass of those is the class d-DIR, consisting of intersection graphs of segments drawn allowing only for d-directions. Since those partition into d interval graphs, we have $\chi \leq \omega d$ there. In an article outside this thesis, with Duraj et al. [Dur+25], we have shown that this function is essentially tight, which is one of the few examples of an exact χ-binding function.

More generally, χ-boundedness has been widely studied for intersection graphs of shapes in the plane, and even in higher dimensions. Asplund and Grünbaum have shown in 1960 that the intersection graphs of axis-parallel rectangles are χ-bounded [AG60]. Perhaps one of the most general results is given by Rok and Walczak [RW19], showing that so-called outerstring graphs are χ-bounded.

Other examples of χ-bounded classes include those with bounded parameters, such as classes of bounded clique-width [BP20], bounded twin-width [PS23, BT25], or even bounded merge-width [BG25]. Until recently, even less was known about the form (minimum) χ-binding functions may take across various χ-bounded classes. To highlight this, Esperet made the provocative conjecture in 2017 that every χ-bounded class is polynomially chi-bounded (that is, f can be a polynomial) [Esp17]. This was recently disproved by Briański et al. [BDW23], generalizing a construction of Carbonero et al. [Car+23].

Forcing trees or cliques

All the directions we explored until now strive to find cliques, the obvious suspect, either as a minor or as a subgraph, to indeed be responsible for high χ. When this is the case, we may of course retrieve any graph as (non-induced) subgraphs of sufficiently large cliques. But in the case where cliques do not appear despite large χ, can we necessarily find other subgraphs? A fundamental result by Erdős [Erd59] is the existence of graphs with arbitrarily large girth and chromatic number (see Theorem 2.3.1). Therefore, the only subgraphs we may expect from large χ are trees, and we will see that this is easily the case through the following folklore lemma.

Lemma 1.1.18. *Let G be a graph such that $\chi(G) \geq k$, then G contains all trees of order k as subgraphs.*

Now, we know that cliques are obstacles to finding induced trees from high chromatic number. This is not the case for graph classes forbidding a given clique but still having high χ. We may then dare to ask for induced copies of trees in such classes, which is what Gyárfás [GST80] and Sumner [Sum81] independently conjectured in what became one of the most famous conjectures on colouring.

Conjecture 1.1.19 (Gyárfás, Sumner '80)**.** *For any tree T and any K_k, any graph with sufficiently large chromatic number contains either T or K_k as an induced subgraph.*

Equivalently, this is saying that for any T, the class of induced-T-free graphs is χ-bounded. The first results of this thesis will deal with analogues of these two statements in the case of directed graphs.

1.2 Towards directed graphs

A *directed graph*, or *digraph*, is a pair $D = (V, A)$, where V is the set of vertices, and a set A of ordered pairs of vertices, called *arcs* or *directed edges*. An *oriented graph* is a digraph that does not contain any *digon*, which are pairs $(x, y), (y, x) \in A$ (directed cycles of length 2). Given a digraph $D = (V, A)$ and a vertex $u \in V$, a vertex $v \in V$ is an out-neighbour of u when $(u, v) \in A$, and v is an in-neighbour when $(v, u) \in A$. When using neighbour without specification, we mean either an out-neighbour or an in-neighbour. Then, the *out-neighbourhood* of u is defined as $N^+(u) = \{v \in V : (u, v) \in A\}$, and its *in-neighbourhood* is $N^-(u) = \{v \in V : (v, u) \in A\}$. The neighbourhood of u is $N(u) = N^+(u) \cup N^-(u)$. In a digraph D, a *source* if all its neighbours are out-neighbours, and a *sink* if all of its neighbours are in-neighbours. We say that a subset $X \subseteq V$ *(out)-dominates* (respectively, *in-dominates*) a subset $Y \subseteq V$ if every $y \in Y$ is the out-neighbour (respectively in-neighbour) of some $x \in X$. Much like graphs, digraphs can also be (di)graphed. While vertices are still represented by points, an arc (u, v) is drawn as an arrow going from u to v. Throughout this thesis, all digraphs are considered up to isomorphism, except in Chapter 8. Although our terminology and definitions are self-contained, we refer the reader to Bang-Jensen and Gutin's book, Digraphs [BG09], for an exhaustive introduction to the topic.

1.2.1 From graphs to digraphs

An oriented graph D may be obtained from a graph G by *orienting* each edge $\{u, v\}$ of G, meaning choosing the ordered pair (u, v) or (v, u). In that case, D is called an *orientation* of G. Conversely, given a digraph D, its *underlying graph* $\mathrm{UG}(D)$ is the undirected graph on the same vertex set, with an edge $\{x, y\}$ whenever (x, y) or (y, x) are arcs of D. This allows for any undirected property to be translated to a directed property, by simply ignoring directions. Most of the time, properties on the underlying graph have little to do with the "directions" of the digraph itself. Usually, inherently directed properties are obtained from undirected ones by relaxing their definition to introduce some kind of asymmetry, as permitted by arcs.

Even though digraphs do not allow for undirected edges, their behaviour can usually be "simulated" by digons. A *bidirected graph* D is a digraph in which every edge is a digon, meaning $(x, y) \in A$ iff $(y, x) \in A$. The bidirected digraph associated to a graph G, denoted $\overleftrightarrow{G}$, is defined on $V(G)$ by replacing every edge with a digon. Bidirected graphs are in bijection with usual graphs, and problems may be expressed equivalently on graphs or the bidirected graph associated to them. In that sense, asking questions about bidirected graphs only is not of interest for digraph theory. Often, given a problem on undirected graphs, the directed analogue coincides with the initial one when considered over bidirected graphs. For this reason, many problems only consider oriented graphs, on which problems differ the most from the undirected version, or at least non-bidirected ones.

1.2.2 Basic digraphs

Let us describe some analogues of the fundamental graphs described in the undirected case. First, we consider orientations of the graphs we already described, to then argue for a more sensible, relaxed analogue in the directed realm.

Orientations of paths and trees

An *oriented path* is the orientation of a path, while an *oriented cycle* is the orientation of a cycle. The natural analogue of a path P_n is the *directed path* $\overrightarrow{P_n}$, defined as the orientation of P_n with arcs $\{(v_i, v_{i+1}) : i < n\}$. In a digraph, a vertex x *reaches* a vertex y if there exists a directed path from x to y, note this yields an asymmetric relation. Now, the *directed cycle* $\overrightarrow{C_n}$ of length n, is the orientation of C_n with arcs $\{(v_i, v_{i+1[n]}) : i \leq n\}$.

There are many ways to orient a tree, but the simplest directed variants of trees are given by *arborescences*. An in-(out-)arborescence is the orientation of a tree T obtained by rooting it at a

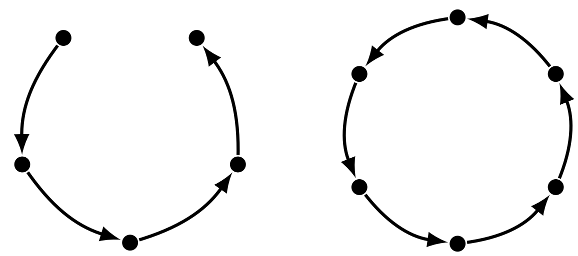


Figure 1.7: The directed 5-path and directed 6-cycle

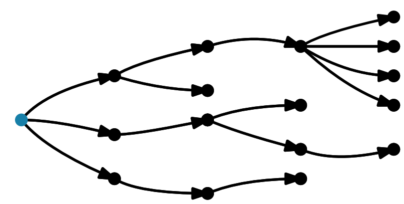


Figure 1.8: An out-arborescence, with its root source shown in blue.

vertex r such that every other vertex has a unique path to (from) r. That is, r will be the unique sink (source), while all leaves will be the sources (sinks), see Figure 1.8.

Independent and acyclic digraphs

While we could generalize forest graphs by taking arbitrary orientations of them, it will be more natural to relax their negative characterization by the absence of cycles. The directed analogue of forests, are digraphs forbidding a directed cycle, which we call *acyclic*, or *DAG* for Directed Acyclic Graph. An ordering $(v_1, \dots, v_n)$ of the vertices of a DAG is a *topological ordering*, or *acyclic ordering*, if whenever (v_i, v_j) is an arc, $i < j$. Remark that any DAG admits an acyclic ordering, which can be obtained by successively pruning its sinks, and conversely acyclic orderings are witnesses for acyclicity. Note any graph G may be oriented as a DAG by choosing an (arbitrary) ordering of its vertices, and orienting edges accordingly. The structure of DAGs may be tamer than general digraphs, but many structural and algorithmic problems remain hard in the class of acyclic digraphs.

While an *independent digraph* corresponds exactly to an independent graph, in many cases, DAGs tend to be the more sensible analogue once more. Indeed, we can understand independent sets to mean that for every pair $x, y \in V$, there is no path between x and y. This is symmetric, so such a path would be back and forth. Then, the relaxed analogue for digraphs allows for one of the directed paths to exist, but not the other. Observe then that this is exactly asking for acyclicity.

Tournaments and semicomplete digraphs

In parallel with independent digraphs, the simplest way to obtain "directed" cliques is to have all possible arcs over V, in both directions, which yields the *bidirected clique* $\overleftrightarrow{K_n}$. A way to relax this is to ask, for each pair $x, y \in V$, that at least one of (x, y) or (y, x) is present, which yields

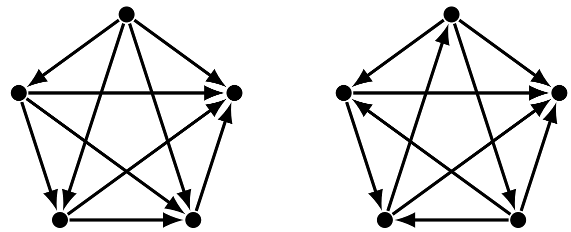


Figure 1.9: The transitive tournament TT_5, and a non-transitive tournament of order 5.

semicomplete digraphs. Note that these are exactly the complements of oriented graphs. Still, the most studied analogues of cliques are *tournaments*, that is, orientations of a clique. The transitive tournament TT_n is the unique acyclic tournament on n vertices, that is, $(v_i, v_j) \in A$ if and only if $i < j$. See Figure 1.9 for examples of tournaments.

1.2.3 Operations and relations

Operations

Most of the time, the operations we deal with for digraphs correspond to the undirected case, such as vertex and arc deletions. Arc augmentations will be our main concern in Chapter 7. A *subdivision* of an arc (x, y) in D removes (x, y), introduces a new vertex z, and adds $(x, z), (z, y)$. An example of directed operation that has no undirected analogues is given by arc *inversions*, which replace (x, y) with (y, x). Generalizing this, the *inversion* of a set of vertices $X \subseteq V(D)$ inverts every arc of $A(D) \cap X^2$. This last operation will be the topic of Chapter 8.

Relations

The notions of subgraphs and induced subgraphs for undirected graphs translate verbatim to digraphs. A *subdigraph* of a digraph D is obtained by considering a subset of its vertices and arcs, while an *induced subdigraph* is obtained by taking a subset of its vertices and preserving all edges between them. When saying "subgraph" for a digraph, we will always mean subdigraph. A *subdivision* of D is any digraph obtained by successive arc subdivisions. An *immersion* of a digraph H within a digraph D is an injective mapping f from $V(H)$ to $V(D)$, along with arc-disjoint paths $f(x)$ to $f(y)$ for every $(x, y) \in V(H)$.

Minors are the reason for much of our understanding of graphs, we may expect a powerful corresponding notion in digraphs. Vertex and edge deletions behave the same as for subgraphs, but note that we have not given an analogue to edge contractions. This is because for many purposes, allowing arbitrary arc contractions seems artificial with respect to the directed properties. Indeed, a contraction may create new directed cycles, meaning strongly connected components are not preserved. Multiple notions have been proposed to overcome this issue, but no analogue emerged as a clear counterpart to the usual minors. In the coming Subsection 1.2.6, we will see two such examples, butterfly and strong minors, as well as their use.

1.2.4 Properties and parameters

Unless stated otherwise, terminology from undirected graphs used for a digraph refers to the graph underlying D. Still, adding an orientation to edges allows for the definition of various properties specific to the directed case.

For a vertex $v \in V(D)$, the two analogues for undirected degree are its *in-degree* $d^-(v) = |N^-(v)|$ and its *out-degree* $d^+(v) = |N^+(v)|$. The *handshaking lemma* for directed graphs is the observation that summing out-(or in-)degrees over all vertices always yields m.

$$\sum_{v \in V(D)} d^+(v) = \sum_{v \in V(D)} d^-(v) = m(D)$$

Then, the *maximum out-degree* of D is $\Delta^+(D) = \max_{v \in V(D)} d^+(v)$, its *maximum in-degree* is $\Delta^-(D) = \max_v d^-(v)$. The *minimum out-degree* of D is $\delta^+(D) = \min_v d^+(v)$, and its *minimum in-degree* is $\delta^-(D) = \min_v d^-(v)$. We also define $\Delta_{max}(D) = \max_v \max(d^-(v), d^+(v))$, $\Delta_{min}(D) = \max_v \min(d^-(v), d^+(v))$, as well as $\delta_{max}(D) = \min_v \max(d^-(v), d^+(v))$ and $\delta_{max}(D) = \min_v \min(d^-(v), d^+(v))$. There are also analogues to degeneracy for digraphs, which will be surveyed in Part II, but let us mention a few important ones. A digraph is k in-(out-)degenerate if all of its subgraphs contain a vertex of in-(out-)degree at most k. Then, it is k-min-degenerate if all of its subgraphs contain a vertex where either the in or the out-degree is at most k. The *directed girth*, or *digirth*, of a digraph is the length of the minimum directed cycle it contains.

1.2.5 Connectivity

Connectivity becomes an asymmetric notion in digraphs, where a vertex x can reach y without the converse being true. The analogue of a graph being connected is given by strong connectivity, which we define below. Then, we will see analogues of Menger's theorem, and problems involving directed connectivity of interest to us.

Strong connectivity

A digraph D is *strongly connected* if for any pair of vertices x, y, there exists a directed path from x to y (thus also from y to x). A *dicut* of D is a partition $(X, V \setminus X)$ of V where no arc is directed from $V \setminus X$ to X. Note that such a cut exists if and only if D is not strongly connected. Then, a *dijoin* of D is a subset $J \subseteq A(D)$ containing an arc across (the bipartition of) each dicut of D. A *strongly connected component*, or *SCC*, of D is a maximal strongly connected induced subgraph. As in the undirected case, these partition the vertices of D. A strongly connected component X is a source (respectively sink) component of D if $(X, V \setminus X)$ (respectively $(V \setminus X, X)$) is a dicut. Given a partition of $V(D)$ into $(X_1, \ldots, X_k)$ inducing the SCCs of D, the *condensation* of D is the acyclic digraph obtained by identifying each SCC into a single vertex, that is $D/(X_1, \ldots, X_k)$.

Note that a graph being connected does not mean it can be oriented to be strongly connected, as witnessed by orientations of paths. The following theorem, due to Robbins, gives a characterization of the graphs for which this is possible.

Theorem 1.2.1 (Robbins '39). *A graph G admits a strongly-connected orientation if and only if it is 2-edge connected.*

This generalizes to strengthenings of strong connectivity, as Nash-Williams showed G admits a "k-arc connected" orientation (see next subsection) if and only if it is $2k$-edge connected [Nas60].

Questions about orienting (undirected) graphs are interesting for their own sake, but differ from our interests here, where we will mostly be looking at altering an already directed graph.

Menger's analogues

The connectivity from a vertex x to a vertex y can be quantified by the number of vertex or edge disjoint *directed* paths from x to y. These define local *vertex connectivity* and local *arc connectivity*, which extend to the whole digraph by taking the minimum value over all pairs. As in the undirected case, connectivity can also be expressed by considering separators and directed cuts, defined as follows. A *directed cut* from x to y is a subset of $A(D)$ the deletion of which yields a digraph with no paths from x to y. We stress the distinction with dicuts, which are merely witnesses of a disconnection from one side of the graph to the rest. Then, note that removing a directed cut from x to y results in a digraph where x and y lie on opposite sides of some dicut. A *(directed) x, y-separator* is a subset of $V(D)$ which, removed, yields a digraph with no paths from x to y. Then, Menger showed that these notions are dual in the same way as in the undirected case.

Theorem 1.2.2 (Menger '27)**.** *Let D be a digraph, and $x, y \in V(D)$ two (non-adjacent) vertices. Then, the size of a minimum directed cut (separator) from x to y is equal to the number of edge-(vertex-)disjoint directed paths from x to y.*

Disconnection problems

Thanks to the duality expressed by Menger's theorem, and since flow problems are inherently directed, finding minimum directed cuts and separators in a digraph D can also be done in polynomial-time via a max-flow algorithm. As motivated in Subsection 1.1.7, we are interested in problems reducing the connectivity between every pair of vertices. Again, if we require directed cuts in both directions for any pair, this is asking for an anticomplete graph as a result, which is not our purpose.

Here, the first interesting variant asks for a directed cut in at least one of the directions for each pair, which yields the following question.

Question 1.2.3. *What is the minimum number of (vertex or arc) deletions rendering a digraph acyclic?*

As in the undirected case, we could relax this to only ask for a quantitative decrease in connectivity, but the question above is already interesting enough. It is already the analogue of breaking 2-connectivity (or cycles), making it a feedback problem, but can also be seen as generalizing undirected deletions to an anticomplete graph. Part II introduces feedback problems in digraphs, asking for few deletions to acyclicity. Then, in Chapter 8 we consider "inversions", which generalize arc deletions, and ask for a minimum number of such operations rendering a digraph acyclic.

Connection problems

Now, what happens if we instead wish to obtain a strongly connected digraph? As in the undirected case, this question is inherently global, and most interesting in the case of arc augmentations.

Question 1.2.4. *What is the minimum number of arc augmentations to make a digraph strongly connected?*

Without more restrictions, we will see that the problem above is not such a hard one. We survey simple cases of this question in Part II, before tackling strong connectivity augmentation for planar digraphs in Chapter 7.

1.2.6 (Lack of) minor structure

While the minor structure theorem (1.1.13) cemented minors as the natural notion for undirected graphs, the plot thickens for minors in digraphs. There are various notions of directed minors, and partial results echoing undirected minor structure, but a universal notion akin to minors seems elusive.

A pressing question for digraphs is to give a satisfactory analogue to treewidth, both in terms of structure, or for algorithmic purposes. Multiple analogues have been proposed, each with different applications. Johnson et al. [Joh+01] introduced the notion of directed treewidth, and they and Reed [Ree97] conjectured the existence of an analogue to the grid minor theorem (1.1.12) for *butterfly minors*. The only operation permitted by those is the contraction of arcs (x, y) for which either x has no other outgoing arc or y has no other incoming arc. This notion is natural as it cannot affect which vertices are reachable from one-another. A directed grid is (roughly) a strongly connected orientation of the undirected grid, with arcs joining both sides to form a cylinder. Again, the directed grid of size k has directed treewidth at least k. The (loose) converse, which is the analogue of the grid minor theorem was only proven 20 years later by Kawarabayashi and Kreutzer [KK15].

Theorem 1.2.5 (Kawarabayashi, Kreutzer '15)**.** *There exists a function f such that any digraph with directed treewidth at least $f(k)$ contains a directed grid of size k as a butterfly minor.*

Of course, the main missing result for digraphs is an analogue of the minor structure theorem (Theorem 1.1.13), which appears out of reach with current notions. Still, Kim and Seymour introduced the notion of *strong minors*, and obtained an analogue for semicomplete digraphs [KS15], which implies the result for tournaments as well.

Theorem 1.2.6 (Kim, Seymour '15)**.** *Any class of semicomplete digraphs closed under strong minors can be defined by a finite set of forbidden strong minors.*

1.2.7 From colouring to dicolouring

The chromatic number $\chi(D)$ of a digraph D refers to the chromatic number of its underlying graph. Colourings have been adapted to digraphs in various ways, but since DAGs are a directed analogue of independent sets, it is natural to relax colourings by asking for partitions into acyclic digraphs. This motivates the following notion, introduced by Neumann-Lara in 1982 [Neu82]. A (proper) *dicolouring* of a digraph D is a partition of $V(D)$ into sets inducing acyclic subgraphs. The classes are also called colours, and each colour class may now contain arcs, but no directed cycle. Then, the *dichromatic number* $\vec{\chi}(D)$ of D is the minimum size of such a partition. A k-*dicritical* digraph is a (subgraph) minimal digraph having dichromatic number k. Observe that there is a one-to-one correspondence between the k-colourings of a graph G and the k-dicolourings of $\overleftrightarrow{G}$, since any digon is a cycle, and must have its endpoints coloured differently. Therefore, $\chi(G) = \vec{\chi}(\overleftrightarrow{G})$, and any question asking to bound χ may be seen as bounding $\vec{\chi}$ for bidirected graphs. Then, it is natural natural to consider stronger versions of the question and ask if they hold for oriented graphs.

The most famous problem about dicolourings is to find an analogue of the four-colour theorem. Since bidirected planar graphs behave the same as planar graphs, they admit both lower and upper

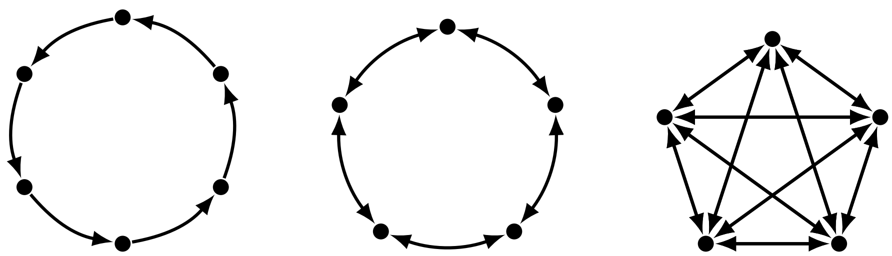


Figure 1.10: Obstructions to Theorem 1.2.8: a directed cycle, a bidirected odd cycle, and a biclique.

bounds of 4. The question becomes more interesting when considering oriented graphs though, and Neumann-Lara conjectured the following in [Neu82].

Conjecture 1.2.7 (Neumann-Lara '82)**.** *Every planar oriented graph is* 2*-dicolourable.*

We will see that getting a bound of 3 can be done by degeneracy arguments. Today, the most general result supporting this conjecture is that it holds for planar oriented graphs forbidding a directed triangle, as shown by Li and Mohar [LM17].

Upper bounds

Any (usual) colouring of a digraph D also yields a dicolouring, so $\vec{\chi}(D) \leq \chi(D)$. This already bounds $\vec{\chi}$ from above by undirected parameters bounding χ, but what about directed ones? The following directed version of Brooks' theorem was first proved by Harutyunyan and Mohar in [HM11]. Aboulker and Aubian gave four new proofs of the result in [AA23].

Theorem 1.2.8 (Harutyunyan and Mohar '12)**.** *Let $D = (V, A)$ be a connected digraph. Then*

$$\vec{\chi}(D) \leq \Delta_{max}(D) + 1$$

and equality holds if and only if D is a directed cycle, a bidirected odd cycle, or a biclique.

The inequality part is easy to see, by colouring vertices one by one with a colour different from either their out-neighbourhood or their in-neighbourhood, which ensures no monochromatic directed cycles appear. Note that considering this result over bidirected graphs yields the original Brooks theorem.

In fact, we even have $\vec{\chi}(D) \leq \Delta_{min}(D) + 1$, by applying the same strategy except we choose a colour different from the smallest neighbourhood. For this notion, Picasarri-Arrieta [Pic24a] strengthened Theorem 1.2.8 and proved that $\vec{\chi}(D) \leq \Delta_{min}$ holds unless D contains the directed join of $\overleftrightarrow{K_r}, \overleftrightarrow{K_s}$ with $r + s = \Delta_m in(D) + 1$.

Similarly to graphs, upper bounds can also be obtained from degeneracy analogues, the following being the strongest.

Observation 1.2.9. *If D is k-min degenerate, then $\vec{\chi}(D) \leq k + 1$.*

Otherwise, we may take a minimal counter-example D and a vertex v with $d^-(v)$ or $d^+(v)$ at most k. Say we are in the first case, we can then colour v with a $k+1$th colour different from those assigned to vertices of $N^-(v)$, which still yields a dicolouring, contradicting the minimality. This is of interest for planar oriented graphs. Indeed, their underlying graph is 5 degenerate, thus a vertex of degree 5 in any subgraph has either $d^- \leq 2$ or $d^+ \leq 2$. That is, they are 2-min-degenerate, which gives an upper bound of 3 for Conjecture 1.2.7.

Lower bounds

The bidirected clique $\overleftrightarrow{K_k}$ is a simple example of a digraph with dichromatic number k. Still, there is no clear analogue of cliques for oriented graphs. Perhaps the most standard construction of oriented graphs achieving high $\vec{\chi}$ is given by starting with a directed triangle and inducting as follows. At any step, we consider a directed triangle, and use two copies D, D' of the oriented graph from the previous step to substitute two vertices of the triangle. Thus, in the resulting oriented graph, D dominates D', while the third vertex of the triangle out-dominates D and in-dominates D'. It is easy to see that each step increases $\vec{\chi}$ by one, but the number of vertices grows exponentially. As for graphs, we can obtain a lower bound by controlling the size of partition classes, which are now acyclic subdigraphs.

Observation 1.2.10. *Let D be a digraph and k be the largest order of an acyclic subdigraph of D, then $\vec{\chi}(D) \geq \frac{n}{k}$.*

1.2.8 Subdigraphs and (di)chromatic number

In this thesis, we seek to understand the relation between digraph structure with respect to both the chromatic number and the dichromatic number. These questions will be surveyed in more depth in the introduction to Part I, but we present two related questions to which we contribute.

Chromatic number

At first glance, there is no reason to expect the chromatic number of the underlying graph to be linked with the oriented graphs it contains. Setting aside the lack of a true directed analogue to minors, any "tight" adaptation of Hadwiger's conjecture (1.1.15) would have to consider all tournaments as outcomes to high chromatic number, which defeats our purpose. We should first ask if any directed substructure is indeed guaranteed at all when χ is large. We will survey the state of this question for weaker notions in Part I, but we are mainly concerned with this question for subdigraphs.

Question 1.2.11. *What subdigraphs are forced to appear in digraphs with large chromatic number?*

Then, the same obstacles of high girth and high chromatic number given by Theorem 2.3.1 apply, so we can only look for orientations of trees. Surprisingly, Burr showed in 1980 that these may all be found, as he showed that any oriented tree of order k is contained in any $(k-1)^2$-chromatic digraph [Bur80]. He conjectured the following holds, which is saying merely doubling the chromatic number required for trees on k vertices should yield all their orientations.

Conjecture 1.2.12 (Burr '80)**.** *Every oriented tree of order k is contained in any $(2k-2)$-chromatic digraph.*

We survey different approaches and results on this conjecture in Part I leading into our sub-quadratic bound in Chapter 3.

Dichromatic number

With the stronger notion of dichromatic number, we should expect to obtain richer substructures from high $\vec{\chi}$. This richness can come either by asking for more structures, or by containment in a stronger sense than what we expect from high χ. While directed minors do not seem to be the natural notion for these questions, it was shown by Aboulker et al. [Abo+19] that high enough $\vec{\chi}$ implies all subdivisions of a given digraph. This will be further exposed in Part I. We know oriented trees appear as subgraphs for high enough $\vec{\chi}$, since $\chi \geq \vec{\chi}$. Rather than trying to ask for more than trees, our concern is to find "sparseness" conditions under which we can induce them.

Question 1.2.13. *When are induced subtrees forced to appear in digraphs with large dichromatic number?*

Any tournament of high $\vec{\chi}$ would stand in our way to finding induced trees, so we should forbid them, which we will see amounts to forbidding the transitive tournament. This motivated Aboulker, Charbit and Naserasr to conjecture the following analogue of the Gyárfás-Sumner conjecture [ACN21] for digraphs.

Conjecture 1.2.14 (Aboulker, Charbit, Naserasr '21)**.** *For any oriented tree T and any k, any digraph with sufficiently large dichromatic number contains either T or TT_k as an induced subdigraph.*

We further motivate this conjecture in Part I, before attacking it in Chapter 4.

1.3 Parameterized complexity

In this section, we introduce the field of parameterized complexity, under which we tackle the otherwise hard (dis)connection problems of Part II. We direct the curious reader to the pioneering book of Downey and Fellows [DF99] and the more modern one by Cygan et al. [Cyg+15].

Facing the growing number of NP-completeness results, the task of computer scientists has been to categorize problems in a binary way: either providing a polynomial (time) algorithm, or showing hardness through reductions. Over the past few decades, parameterized complexity emerged as a way to overcome the exponential complexity of NP-complete problems. While usual complexity theory measures the hardness of a problem relative only to the size of instances, this is often an overshoot, as it is driven by worst-case complexity. Indeed, many NP-complete problems become efficiently solvable, provided the inputs are restricted, such as trees, or graphs with bounded degree, so we may ask for more.

Question 1.3.1. *For any algorithmic problem, what are the properties of instances allowing for efficient resolution?*

Parameterized complexity looks at isolating such properties in a way that we can measure quantitatively. Towards this, it considers each instance I along with an inherent parameter k, and factors it into the complexity measure, expressing it as some $f(k, |I|)$. As a consequence, even hard algorithmic problems may admit efficient algorithms in terms of $|I|$, as long as instances are bounded in terms of the parameter. Intuitively, the goal is to shift the exponential blow-up arising in NP-hard algorithms to a term dependent only on k. We stress that the adequate parameter may differ from one problem to another.

Parameterized problems

In the following, we consider a *parameterized problem*, which is a language $\mathcal{L}$, consisting of instances (I, k). Here, I is the instance, in our case the (di)graph, and k is the parameter. In this thesis, those will be stated in the following form:

PROBLEM **Parameter:** k **Input:** Instance (I, k) **Question:** Is (I, k) positive? That is, does it belong to $\mathcal{L}$?

We will often only state the parameterized version, the usual decision version merely drops the parameter in the above and fixes k. The routine parameter k to consider is size of the solution we are looking for, which is often a sensible choice. Solution size will always be the implied parameter when none is specified, and prefixing an unparameterized problem with "k-" means considering the (parameterized) decision problem asking whether I admits a solution of size k. Otherwise, k may be a parameter of the graph at hand, such as its maximal degree, or its treewidth. It may even be a tuple (k, k') of multiple parameters, in which case we also say it is parameterized by k and k'.

1.3.1 Fixed-parameter tractability and XP

The most efficient class of algorithms in parameterized complexity are those which are *fixed-parameter tractable*, or *FPT*. They are those for which deciding an instance (I, k) can be done in time $f(k)|I|^{O(1)}$. Then, we say the problem is FPT with respect to k. When this is not possible, we may look for *XP algorithms*, running in time $|I|^{f(k)}$ instead. Note that FPT implies XP. In both cases, f is allowed to be any computable function, the takeaway being that fixing k yields a polynomial-time algorithm across all instances with parameter k. Still, FPT is more desirable than XP, as the polynomial exponent is uniform across all parameters.

Treewidth is one of the first parameters which emerged with applications to FPT, even if its introduction was motivated by minor structure. As a measure of structural "tree-likeness", it is natural to expect polynomial algorithms for many problems in graphs with bounded treewidth. The following (meta-)theorem of Courcelle confirmed this belief in a very universal sense [Cou90].

Theorem 1.3.2 (Courcelle '90)**.** *Any problem expressible in Monadic Second Order Logic is FPT parameterized by treewidth.*

This is a major result lying at the intersection of logic, algorithms and graph structure. The majority of algorithmic problems on graphs fall within its scope, and thus there are few problems which we cannot solve when treewidth is bounded.

Still, none of the directed analogues of treewidth enables us to extend Courcelle's theorem to digraphs. In general, much less is known about structural properties of digraphs making problems easier. Of course, the treewidth of the underlying graph can be useful, but when it comes to directed notions, efficient algorithms are often restricted to DAGs or tournaments. In Chapter 8, we rely on tournament structure, but all problems considered in Part II are parameterized by solution size (at least).

1.3.2 Kernels

Kernelization is a natural way to design FPT algorithms by applying preprocessing rules to produce an equivalent, but smaller instance. In the parameterized realm, smallness means that the output size depends only on k, but let us formalize this. Given a parameterized problem $\mathcal{L}$, a *kernelization algorithm* is a polynomial-time algorithm taking an instance (I, k), and producing an instance (I', k'), the *kernel*, such that:

- $(I, k) \in \mathcal{L}$ if and only if $(I', k') \in \mathcal{L}$,
- $|I'| \leq g(k)$ for some computable function f.

Most often, the term kernel will actually refer to the kernelization algorithm, and we may say "the problem admits a kernel". Now, if such a kernel exists, then the output instance (I', k') may be solved by any brute-force approach yielding a complexity of the form $h(g(k))$ for some g. This means that the initial problem is solvable in time $n^{O(1)} + h(g(k))$, yielding fixed-parameter tractability.

In fact, being FPT and admitting a kernel are equivalent notions. Still, a stronger requirement is to ask for a *polynomial kernel*, which additionally imposes that the size $g(k)$ grows polynomially in k. There is an extensive framework called *compositionality*, giving (conditional) non-existence results for polynomial kernels, or even lower bounds on their exponent.

1.3.3 Hardness

Sadly, not all problems are FPT, or even XP, and there exist parameterized hardness classes supported by largely believed assumptions. A parameterized problem is para-NP-hard if it is NP-hard even for a constant parameter. A typical example of this is k-colourability parameterized by k, as 3-colourability is already NP-hard. These are the hardest problems here, where parameterization had no effect, and unless P=NP, they cannot admit XP algorithms.

It is known that FPT is strictly contained in XP, so we would like to be able to tell whenever some problems, which are in XP, are not likely to contain FPT algorithms. The corresponding notion of hardness is given by W[1]-hardness, which, if shown by a reduction from another W[1]-hard problem, rules out an FPT algorithm under the assumption that coNP $\subsetneq$ NP/poly. We will not give the definition of W[1] here, as it is enough to explicit the reduction required to show a problem is W[1]-hard.

A *fixed-parameter reduction* from a problem $\mathcal{L}$ to a problem $\mathcal{L}'$ takes an instance (I, k) for $\mathcal{L}$ and outputs, in FPT time $(f(k)|I|^{O(1)})$, an instance (I', k') for $\mathcal{L}'$ such that:

- $(I, k) \in \mathcal{L}$ if and only if $(I', k') \in \mathcal{L}'$, that is, the two instances are equivalent,
- $k' \leq g(k)$ for a function g.

Then, giving a fixed-parameter reduction from a W[1]-hard problem to some problem $\mathcal{L}$ shows that $\mathcal{L}$ is also W[1]-hard.

1.4 Overview of the results

In a broad sense, all our questions can be framed in two directions to overcome the structural complexity in directed graphs, reflected in the two parts of this thesis.

The first direction embraces this complexity, understood in terms of (di)chromatic number, and looks at describing the structures emerging from it. This will be the subject of the three chapters of Part I.

Question 1.4.1. *What are the (induced) subdigraphs of digraphs with high (di)chromatic number?*

The second direction seeks to control it, in terms of connectivity, by breaking or forming global connections through local operations. We will be concerned with these two questions in the two chapters of Part II.

Question 1.4.2. *How to strongly connect a digraph by adding arcs? How to disconnect it by removing or inverting arcs?*

1.4.1 Subdigraphs in colourful digraphs

We begin Part I with an introduction to the topic of finding substructures in graphs and digraphs, from degrees to chromatic number. Then, we focus on our interest in subdigraphs forced by large (di)chromatic number. We present previous approaches and techniques, before moving to our three contributions to this subject.

Oriented trees from large chromatic number

In Chapter 3, we give a subquadratic bound for Burr's conjecture (1.2.12), by showing digraphs with chromatic number $O(k\sqrt{k})$ contain all oriented trees of order k. Along the way, we improve the result for more specific classes of oriented trees such as b-block paths and arborescences.

These results are based on joint work with Stéphane Bessy and Daniel Gonçalves. They were presented at the WG 2024 conference and have been submitted to a journal.

- Stéphane Bessy, Daniel Gonçalves, and Amadeus Reinald. "Oriented Trees in O(K√k)-Chromatic Digraphs, a Subquadratic Bound for Burr's Conjecture". In: *Graph-Theoretic Concepts in Computer Science: 50th International Workshop, WG 2024, Gozd Martuljek, Slovenia, June 19–21, 2024, Revised Selected Papers*. Berlin, Heidelberg: Springer-Verlag, June 2024, pp. 79–91. DOI: 10.1007/978-3-031-75409-8_6

Induced orientations of P_4 from large dichromatic number

In Chapter 4, we are interested in Conjecture 2.6.3, the directed analogue of the Gyárfás-Sumner conjecture proposed by Aboulker, Charbit and Naserasr. We prove the conjecture for all orientations of P_4 in a unified approach, solving the conjecture for directed and antidirected orientations.

These results are based on joint work with Linda Cook, Tomas Masarik, Marcin Pilipczuk, and Ueverton S. Souza. They were presented at Eurocomb '23, and lead to the following journal publication:

- Linda Cook, Tomáš Masařík, Marcin Pilipczuk, Amadeus Reinald, and Uéverton S. Souza. "Proving a Directed Analogue of the Gyárfás-Sumner Conjecture for Orientations of P4". In: *The Electronic Journal of Combinatorics* (Sept. 2023), P3.36–P3.36. DOI: 10.37236/11538

Brooks' theorem for bivariable degeneracy

In Chapter 5, we introduce a new colouring variant partitioning into classes constrained by *bivariable degeneracy*, which generalizes most previous colouring notions. For this, we show an extension of Brooks' theorem ensuring certain structures are forced when the digraph requires many colours. We provide a linear time algorithm producing the associated colouring, or outputting an obstruction, improving on previously known algorithms for more restricted analogues.

These results are based on joint work with Daniel Gonçalves and Lucas Picasarri-Arrieta, leading to the following journal publication:

- Daniel Gonçalves, Lucas Picasarri-Arrieta, and Amadeus Reinald. "Brooks-Type Colourings of Digraphs in Linear Time". In: *Journal of Graph Theory* 110.4 (2025), pp. 496–513. DOI: `10.1002/jgt.23266`

1.4.2 (Dis)connecting digraphs

Part II studies (parameterized) algorithmic connectivity problems aiming at making a digraph strongly connected, or acyclic. We begin in Chapter 6 by introducing connectivity augmentation problems, and analogue problems constrained by planarity, towards plane strong connectivity augmentation. In the same chapter, we survey structural and algorithmic disconnection questions in the form of feedback problems, before moving to inversions in digraphs. In the two parts following the introduction, we present our contributions to each of these (opposing) problems in digraphs.

Augmenting a planar digraph to strong connectivity

In Chapter 7, we consider the problem of making a plane oriented graph strongly connected while remaining oriented and planar. We show that this arc augmentation problem is hard even without restrictions on the number of arcs, and give an FPT algorithm parameterized by the number of arcs in a sought solution.

These results are based on joint work with Stéphane Bessy, Daniel Gonçalves, and Dimitrios Thilikos. They are in preparation for submission to both a conference and a journal. They are in preparation for submission, with the latest version appearing in the following preprint:

- Stéphane Bessy, Daniel Gonçalves, Amadeus Reinald, and Dimitrios M. Thilikos. *Plane Strong Connectivity Augmentation*. Dec. 2025. DOI: `10.48550/arXiv.2512.17904`. arXiv: `2512.17904 [math]`

Disconnecting a digraph through bounded-size inversions

In Chapter 8, we consider inversions of bounded size in digraphs. We show bounds on the maximal number of inversions rendering a digraph acyclic. Then, we show that the problem of deciding if a tournament can be made transitive by k inversions of size at most p admits a polynomial kernel in (k, p), as well as a $W[1]$-hardness result in general digraphs.

These results are based on joint work with Jørgen Bang-Jensen, Frédéric Havet, Florian Hörsch, Clément Rambaud and Caroline Silva. They are in preparation for submission, with the latest version appearing in the following preprint:

- Jørgen Bang-Jensen, Frédéric Havet, Florian Hörsch, Clément Rambaud, Amadeus Reinald, and Caroline Silva. *Making an Oriented Graph Acyclic Using Inversions of Bounded or Prescribed Size*. Nov. 2025. DOI: `10.48550/arXiv.2511.22562`. arXiv: `2511.22562 [math]`

Part I

Subdigraphs in colourful digraphs

Chapter 2

Introduction: looking for sub(di)graphs

This part of the thesis investigates subgraphs of (di)graphs with large (di)chromatic number. We survey numerous variants of these questions, situating our contributions in this vast subfield of structural (di)graph theory. In particular, we investigate subdigraphs of digraphs with large chromatic number in Section 2.5, presenting our results on Burr's conjecture from Chapter 3. Then, we study the problem of induced subdigraphs obtainable from large dichromatic number, and motivate the directed extension of the Gyárfás-Sumner conjecture in Section 2.6, towards our proof for orientations of P_4 in Chapter 4. Finally, in Section 2.7 we look at the extreme case of these questions, when chromatic number exceeds the degree, paving the way to our generalization of Brooks's theorem shown in Chapter 5.

2.1 Substructure in graphs

Many threads in graph theory can be abstracted as the search for conditions forcing the presence of a given substructure in a graph. This leaves room for interpretation on both what these conditions are, and what we understand by containment. The conditions one usually imposes are in the form of an invariant being sufficiently large, this can be the number of edges, degrees, or the chromatic number. Then, the sought substructure may either be a specific graph or a member of a class, and its containment can be understood in terms of any graph relation. Equivalently, this question can be framed as asking to bound some invariant among graphs forbidding a certain structure.

Extremal graph theory is riddled with such questions, usually describing the behaviour of graphs with many vertices or edges. It is concerned with characterizing thresholds for the presence of a given structure, but also which graphs attain them. With such universal parameters, questions in this field typically look for the existence of cliques and stable sets. The well-known Turán theorem gives an upper bound on the number of edges of an n-vertex graph forbidding K_r [Tur41]. On the other hand, Ramsey's theorem ensures that a sufficiently large graph will contain either a large clique or a large independent set [Ram09], although the bound grows exponentially in n [Cam+].

We are concerned with questions of a more structural nature, looking for conditions under which a specific subgraph H appears (equivalently, the properties of graphs forbidding H). The most natural problems of this taste, some conjectured decades ago, are still far from being understood. A notable conjecture formulated in 1989 by Erdős and Hajnal [EH89], states that for any fixed graph H, the bound for Ramsey's theorem among induced-H-free graphs should be polynomial. It stands as one of the most famous structural conjectures in graph theory, with only a few cases known. Interestingly, the Erdős-Hajnal conjecture was shown to be equivalent to a question about dicolouring tournaments by Alon, Pach and Solymosi [APS01]. Their equivalent question states that for any fixed tournament H, there exists an $\epsilon > 0$ such that any tournament T such that $\vec{\chi}(T) > cn^{1-\epsilon}$ contains H.

Substructures from chromatic numbers

In this part of the thesis, we seek to understand both colourings and dicolourings by extracting subgraphs from digraphs with large (di)chromatic number. We motivate many variants of this question, surveying known results and conjectures, and introducing our contributions along the way. For undirected graphs, where minor structure is well understood, the most famous question about substructure from χ is Hadwiger's conjecture (1.1.15). In the directed case, the lack of a satisfying analogue to minors has led researchers to look for substructures rather in terms of (induced) subdigraphs, and their subdivisions. These questions already have a lot of history for undirected graphs, which we survey, notably through the Gyárfás-Sumner conjecture. Our main interest is with subdigraphs and induced subdigraphs forced by high (di)chromatic number. We have already seen these can at best be (oriented) trees, which is also the case for dichromatic number. We refer the interested reader to the survey of Stein [Ste24] for a more exhaustive insight into tree containment. The only exception to this is for (analogues of) Brooks's theorem, where the chromatic number is tied to the maximal degree.

We begin in Section 2.2 by surveying sub(di)graphs already known to be forced by weaker parameters: high degrees and oriented variants. Our question in the undirected case, looking for subgraphs from high chromatic number, will be surveyed in Section 2.3. Surprisingly, digraphs with large (underlying) χ are also forced to contain certain subdigraphs. As we will see in Section 2.4, this is more easily observed when restricting host digraphs, for instance to tournaments or DAGs.

We consider general digraphs in Section 2.5, motivating Burr's conjecture, and present the results towards our subquadratic bound Chapter 3. Afterwards, in Section 2.6, we strengthen the condition of high χ to high $\overrightarrow{\chi}$, which will allow us to ask for induced subdigraphs. This motivates our results on a directed analogue of the Gyárfás-Sumner conjecture in Chapter 4. We then consider the extreme cases of these questions in Section 2.7, in the form of analogues of Brooks's theorem for various colouring notions. That is, we investigate the sub(di)graphs guaranteed when the chromatic number exceeds (analogues of) the maximal degree, which will be the question of Chapter 5.

2.2 Sub(di)graphs from high degrees

In this section, we investigate subgraphs and subdigraphs implied by high (variants of) degree, which will also imply those appear from large χ or $\overrightarrow{\chi}$. Indeed, for undirected graphs $\chi \geq k$ implies a subgraph with $\delta \geq k-1$, thus also $\mathrm{ad} \geq k-1$. Similarly for digraphs, $\overrightarrow{\chi} \geq k$ implies $\delta^-, \delta^+ \geq k-1$ (on top of $\overrightarrow{\chi} \geq \chi$). We first look at the undirected case of this problem in Subsection 2.2.1, proceeding with digraphs in Subsection 2.2.2. At the end of each subsection, we discuss asymptotic questions dealing with forcing a larger class of (di)graphs as subdivisions.

2.2.1 Subgraphs

Here, we discuss the structures contained in undirected graphs with high minimum or average degree, which must therefore also appear when χ is large.

Subtrees

If a graph has sufficiently large degree(s), what (induced) subgraphs does it necessarily contain? Maximal degree is of little interest here as stars are exactly the graphs which it can guarantee, so we should look at high minimum degree δ and average degree ad instead. These two questions are very tied, on the one hand because $\delta \leq \mathrm{ad}$, and on the other because Observation 1.1.1 allows us to extract a subgraph with minimum degree $\mathrm{ad}/2$. In particular, asymptotic considerations are equivalent.

A first obstacle in this quest is the existence of r-regular graphs with girth $g+1$, for any values of r and g, initially due to Biggs [BLW86]. This means that both minimum and average degree cannot guarantee more than trees as subgraphs.

The clique of order k shows that minimum degree $k-1$ cannot guarantee trees of order larger than k, nor any induced tree of order more than 2. The latter observation completely settles the question for induced trees, and we focus on trees as subgraphs. It is easy to see that this degree condition for subtrees is indeed sufficient, as we show in the following folklore result.

Lemma 2.2.1. *Let G be a (non-empty) graph such that $\delta(G) \geq k-1$, then G contains all trees of order k.*

Proof. We fix G, and show this by induction on the order of the tree T, the case $|V(T)| = 1$ being clear. Assume then the result holds for orders less than k, and consider T of order k. Let T' be a subtree of T obtained by removing any leaf, and let u be its parent in T'. By induction, we may find a copy of T' in G. The vertex corresponding to u has degree at least $k-1$ in G, while $|V(T') - u| < k-1$, so u admits at least one neighbour distinct from T', yielding T. □

Note that the latter proof allows us to root the copy of T at any vertex of G, which is not typically the case for other notions. In fact, this can be strengthened by a factor of 2 for paths, as shown by Erdős and Gallai [EG59].

Theorem 2.2.2 (Erdös, Gallai '59)**.** *Any connected graph of order $k+1$ with $\delta(G) \geq \frac{k}{2}$ contains a k-path.*

Minimum degree allows us to find a fixed subtree rooted in any vertex of the host graph. This is not possible when only average degree is large, but we may still look for a tree somewhere. If $\mathrm{ad}(G) \geq 2k$, the successive removal of vertices with degree less than k yields a subgraph with minimum degree k, giving double the bound of Lemma 2.2.1. Still, a famous conjecture by Erdős and Sós posits that the correct bound should in fact be the same.

Conjecture 2.2.3 (Erdős, Sós '63)**.** *Any graph G such that $\mathrm{ad}(G) > (k-2)$ contains every tree of order k.*

The resolution of the conjecture for sufficiently large k was announced by Ajtai, Komlós, Simonovits, and Szemerédi, but has yet to be published.

More than trees

Under what understanding of containment can we force a subgraph H that is not a tree (or forest)? To counter the obstacle of graphs with high girth and degree, we ask for subdivision of H instead. The question of which graphs are (asymptotically) enforcible by high (minimum or average) degree has been completely resolved by Mader, showing H can be taken as any graph [Mad72].

Theorem 2.2.4 (Mader '72)**.** *For every k, there exists an $f(k)$ such that every graph with minimum degree at least $f(k)$ contains a subdivision of K_k.*

Bollobás and Thomason [BT98], as well as Komlós and Szemerédi [KS96] showed that there exists a (universal) constant C such that f can be taken as Ck^2.

A stronger requirement is to look for induced subdivisions, when provided some kind of sparsity constraint avoiding the obvious counter-examples. If sparsity is imposed by forbidding a clique as a subgraph, the fact that bicliques have arbitrarily large average degree means they would be the only kind of graphs which we can force (even as a subdivision). Still, there exist graphs with arbitrarily large average degree forbidding $K_{2,2}$, so bicliques cannot be the only outcome to high average degree in K_t-free graphs. Now, when we forbid a fixed biclique, Kühn and Osthus show any induced subdivision may be obtained [KO04].

Theorem 2.2.5 (Kühn, Osthus '04)**.** *For any graph H, and for every integer k, there exists $f(k, H)$ such that every graph of average degree at least $f(k, H)$ contains either $K_{k,k}$ as a subgraph, or an induced subdivision of H.*

2.2.2 Directed subgraphs

Now, we discuss directed substructures of digraphs when analogues of minimum degree are high, as well as when undirected average degree is.

Oriented subtrees

We should not expect imposing high directed variants of degree to force more than oriented trees. This can be seen as a consequence of $\overrightarrow{\chi}$-critical graphs having $\overrightarrow{\chi}$ lower than the usual directed degree variants, and the existence of digraphs with high $\overrightarrow{\chi}$ and (undirected) girth, due to Harutyunyan and Mohar [HM12]. Let us first consider the least restrictive analogues of minimum degree, minimum out and in degrees. For these, the greedy approach of Lemma 2.2.1 shows that $\delta^+ \geq k-1$ implies an arborescence of order k (and symmetrically for δ^-). We should now ask whether (sufficiently) large minimum out-degree can force more than out-arborescences.

Question 2.2.6. *What are the oriented subtrees of digraphs with sufficiently large minimum out-degree?*

A positive answer was known for in-stars as a consequence of their containment as subdivisions shown in [Abo+19]. Still, the actual question for subgraphs had received little attention until recently, when Hons et al. [Hon+25] showed that any oriented tree T satisfying the above must be "grounded". They define grounded orientations as those where vertices with in-degree at least 2 are at the "same level" of a natural layering. They conjectured that the converse holds as well, in their KAMAK tree conjecture, which was just proven by Christoph and Steiner [CS25].

The above shows that high out (or in) minimum degree alone does not behave analogously to high undirected degree for constructing oriented trees. Still, this analogy can be obtained through the stronger *semidegree*, defined for any v as $\delta^0(v) = \min(\delta^-(v), \delta^+(v))$. Then, it is clear that asking for minimum semidegree at least $k-1$ greedily produces any orientation of a tree of order k. Weakening this condition, Stein recently proposed the following analogue of Theorem 2.2.2.

Conjecture 2.2.7 (Stein [Ste20])**.** *Any oriented graph such that* $\min(\delta^+, \delta^-) > \frac{k}{2}$ *contains every oriented path of order* k*.*

Today, this is known to hold only for $\overrightarrow{P_k}$ by an old result of Jackson [Jac81]. And for 2 block paths, a bound of $\frac{3k}{4}$ was recently obtained by Penev et al. [Pen+25].

Continuing the path laid in the undirected case, note that as a (relative) measure of the number of arcs, average degree is as sensible of a parameter for digraphs as it is for graphs. Can we then ask for oriented trees when only the (undirected) average degree is high? This is already quite restricted, as $K_{t,t}$ has average degree t, yet orienting it with all arcs going from the first independent set to the second avoids any $\overrightarrow{P_3}$. That is, we may only ask for antidirected trees, in which every vertex is either a source or a sink of the orientation. The mere fact that sufficiently large average degree guarantees any antidirected tree was conjectured by Erdős, and shown by Graham [Gra70]. Burr showed that $ad \geq 8(k-1)$ ensures all antidirected trees of order k [Bur80], and that average degree at least $2(k-2)$ is necessary. We sketch the sufficient condition below, as it is one of the few cases of a linear bound for Burr's conjecture.

Theorem 2.2.8. *If a digraph D satisfies* $\mathrm{ad}(D) \geq 8(k-1)$*, then it contains all antidirected trees of order* k*.*

Proof. Consider any bipartition (A, B) of $V(D)$, and, as long as there exists a vertex $v \in A$ ($v \in B$) such that $d_B(v) < d_A(v)$ ($d_A(v) < d_B(v)$), swap v to the other class. At each step, the number of edges across (A, B) increases, meaning the process ends. Eventually, every vertex admits at least as many neighbours in the other class of the bipartition as it does in its own. In turn, $m(A) = \frac{1}{2}\sum_{v \in A} d_A(v) \leq \frac{1}{2}\sum_{v \in A} d_B(v) = m(A,B)/2$, and $m(B) \leq m(A,B)/2$ symmetrically.

Therefore, $2m(A,B) \geq m(D) \geq 4(k-1)n$, so $m(A,B) \geq 2(k-1)n$, and we take D' to be the bipartite digraph on $V(D)$ obtained by considering only $e(A,B)$.

Then, we may assume that at least half of the arcs are directed from A to B, without loss of generality. Take D'' to be the subdigraph of D' obtained by only considering those arcs. Then, $m(D'') \geq (k-1)n$, so $\mathrm{ad}(D'') \geq 2(k-1)$, and we use Observation 1.1.1 to obtain a subdigraph H with minimum degree at least $(k-1)$. Now, consider any antidirected oriented tree T of order k, and take s to be any source of T. Then, observe that to build T in H, it suffices to find an undirected copy of T where the vertex corresponding to s lies in A. Since $\delta(H) \geq k-1$, this follows from Lemma 2.2.1. □

More recently, Addario-Berry et al. [Add+13] gave a slight improvement to Theorem 2.2.8, and conjectured his necessary condition is also sufficient.

Conjecture 2.2.9 (Addario-Berry et al. '13)**.** *If a digraph D is such that $\mathrm{ad}(D) > 2(k-2)$, then D contains every antidirected tree of order k.*

More than oriented trees

As in the undirected case, we can now relax the containment condition and look for subdivisions of digraphs that are not necessarily oriented trees. In this vein, Mader conjectured the following analogue of Theorem 2.2.4 for transitive tournaments in [Mad85].

Conjecture 2.2.10 (Mader '85)**.** *For every k, there exists an $f(k)$ such that every digraph with minimum out-degree at least $f(k)$ contains a subdivision of TT_k.*

Although this remains wide open for $k \geq 5$, Lochet showed such a function exists for when asking for immersions of TT_k [Loc19]. Here, we can even ask for subdivisions of graphs which are not DAGs, as out-degree at least k yields a directed cycle of length at least $k+1$ (say by considering a maximal directed path). Little is known about the shape of general digraphs appearing as subdivisions for sufficiently large out-degree. A notable example is the case of finding disjoint directed cycles, which has received some attention through a conjecture of Bermond and Thomassen [BT81] that $\delta^+ \geq 2k-1$ implies k disjoint directed cycles. While the conjecture is only known in full for $k = 2$ [Tho83] and $k = 3$ [LPS09], linear ($\delta^+ \geq ck$) bounds have been obtained [Alo96, Buc18, BBT14, LPS09].

Going in the other direction, it is natural to expect that if δ^+ is very large, say tied to n, a digraph will contain some short directed cycle. By the handshaking lemma, $\delta^+ < n/2$ for oriented graphs, so already $\delta^+ \geq \frac{n}{2}$ yields a $\overrightarrow{C_2}$. Generalizing this, Caccetta and Haggkvist posed the following notorious conjecture [CH78].

Conjecture 2.2.11 (Caccetta, Haggkvist '78)**.** *Every digraph with minimum out-degree at least r has a directed cycle of length at most $\lceil \frac{n}{r} \rceil$*

Despite receiving a lot of attention, this remains open in the general case, which is only known for $r \leq \sqrt{n/2}$ by Shen [She02]. Relaxations of the conjecture have also been shown when allowing multiplicative ([HKN17]) or additive ([She02]) constants, see also the survey by Sullivan [Sul06].

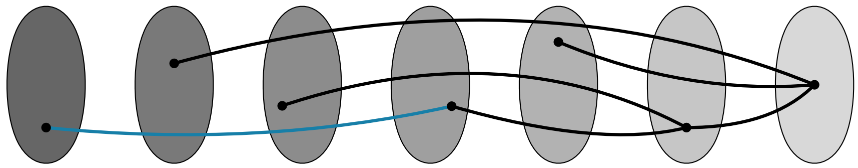


Figure 2.1: Building a tree of order 7 when $\chi = 7$. The colouring shown is obtained by "stacking" successive maximal independent sets. A tree contained in the last 6 classes can be extendend using the first, as shown in blue.

2.3 Subgraphs from high chromatic number

In this section, we only consider undirected graphs, and look for subgraphs forced by high chromatic number. Recall that when asking strictly for (induced) subgraphs, these can only be trees, because of the following result by Erdős [Erd59].

Theorem 2.3.1 (Erdős '59)**.** *There exist graphs of arbitrarily large girth and chromatic number.*

A fortiori, this also restricts our later questions for induced subgraphs, as well as oriented subdigraphs, to questions about (oriented) trees.

2.3.1 Subtrees

The clique of order k shows that the largest tree we may hope to find when $\chi = k$ is one of order k. We recall the following folklore lemma saying that this is always possible, completely settling our question for subgraphs.

Lemma 2.3.2. *Let G be a graph such that $\chi(G) \geq k$, then G contains all trees of order k as subgraphs.*

We show this in two ways, which generalize to various applications for related digraph questions.

The first way to see this is true is that if a graph G satisfies $\chi(G) \geq k$, it contains a subgraph with minimum degree $k-1$. This can be obtained by taking a k-critical subgraph of G, where removing a vertex with degree less than k would contradict minimality. Then, the existence of any tree of order k is guaranteed by Lemma 2.2.1.

The following technique uses χ to build a "stacked" colouring, maximizing successive classes. This witnesses enough structure between colours to build "multicoloured" trees, as we show below, see Figure 2.1.

Proof of Lemma 2.3.2. We proceed by induction on k, with the case $k = 1$ being trivial. Now, assume the result holds for $k-1$, and let us consider G with $\chi(G) \geq k$. Let I be a maximal independent set of G, thus also a dominating set, and take $G' = G - I$, yielding $\chi(G') \geq k-1$. Consider any tree T of order k, and let T' any subtree of T, say by removing a leaf with parent u. Then, by induction T' is conained in G', and since I is dominating, a neighbour may be appended to u from I to build T. □

2.3.2 Induced trees: the Gyárfás-Sumner conjecture

What happens if instead of asking for high χ to force a mere subgraph, we want its copy to be *induced*? Again, we can only hope for trees, and because of cliques, this can only be asked within sparse classes, which for our purpose are those forbidding a given subgraph. Recall from Theorem 2.2.5, that to obtain interesting induced subgraphs from sufficiently high degree, forbidding a biclique guarantees subdivided copies of any graph. This implies the result for high χ in turn. Here, we are interested in a more general class of graphs, only forbidding a large clique, meaning arbitrary bicliques may appear, and high degree will not be sufficient to guarantee an induced tree. Of course, cliques are an obstacle to high χ, but we have seen they are not the only one. Therefore, our question can be formulated as asymptotically asking for two outcomes to large chromatic number: either a large clique, or a fixed induced tree. Recall this is exactly the Gyárfás-Sumner conjecture (2.3.3), which we restate here in terms of χ-boundedness, as we later introduce directed variants of this concept.

Conjecture 2.3.3 (Gyárfás-Sumner '80)**.** *For every tree T, the class of (induced) T-free graphs is χ-bounded.*

This conjecture is sometimes stated by replacing trees with forests, which is equivalent since forests are induced subgraphs of trees. In his seminal paper, Gyárfás showed this to be true for paths [Gyá87].

Theorem 2.3.4 (Gyárfás '80)**.** *For any $n \in \mathbb{N}$, the class of P_n-free graphs is χ-bounded by the function $f(\omega) = (n-1)^{\omega-1}$.*

The elegant technique used by Gyárfás to prove Theorem 2.3.4 has since been generalized to other contexts, and is now dubbed the "Gyárfás path argument". Other than paths, the most general graphs for which the conjecture is known are trees of radius two, and those obtained from them by 1-subdivisions of some vertices incident to a root, thanks to Scott and Seymour [SS20b]. We refer to the survey of Scott and Seymour [SS20a] for a recent overview on the state of the conjecture.

Considering the largely open status of the conjecture, it would be interesting to obtain weakenings either by asking for stronger requirements than high χ, or by allowing a weaker notion of containment. On the first front, we later introduce Conjecture 2.6.5, substituting large χ for large $\overrightarrow{\chi}$ (in an orientation of G). On the second front, Scott proved a weakening of Conjecture 2.3.3 when asking for copies of T only as induced subdivisions [Sco97].

Theorem 2.3.5 (Scott '97)**.** *For every tree T and integer k, there exists an integer $f(k,T)$ for which every graph G such that $\chi(G) \geq f(k,T)$ contains either K_k, or an induced sudivision of T.*

The Gyárfás-Sumner conjecture characterizes pairs $\{K,T\}$ of a clique and a tree as bounding the chromatic number when forbidden. Extending this, we may ask for a characterization of all such sets $\mathcal{F}$, called *heroic*, such that high chromatic number implies one induced copy of an element in $\mathcal{F}$. Assuming the conjecture for $\{K,T\}$ is indeed true, Chudnovsky et al. [CS14] characterized all such sets. This type of extension is also the origin of directed analogues to the Gyárfás-Sumner conjecture, which we introduce in Section 2.6.

2.3.3 More than trees

More permissive notions of containment such as (induced) subdivisions and minors circumvent the obstacles for finding cycles given by Theorem 2.3.1. For minors, this is exactly Hadwiger's conjecture 1.1.15, stating that $\chi \geq k$ forces K_k, thus any graph of order k as a minor. Hajós asked whether Hadwiger's conjecture could be strengthened by asking for subdivisions of K_k, but this was ruled out by Catlin [Cat79]. Still, we know as a consequence of Theorem 2.2.4, that a non-induced subdivision of K_k appears for large enough χ, as this implies high minimum degree subgraphs. If we want the subdivisions to be induced, Scott conjectured an extension of his Theorem 2.3.5: any graph H may appear as an induced subdivision of a K_k-free graph for sufficiently large χ. This was disproven by Pawlik et al. by showing that a certain family of string graphs, not containing induced subdivisions of $K_{3,3}$, is not χ-bounded [Paw+14].

2.4 Subdigraphs from chromatic number: specific classes

In this section, we are concerned with forcing subdigraphs from high (undirected) chromatic number, within tournaments and DAGs. Towards generalizing results to all digraphs, constrained by Theorem 2.3.1, we take a particular interest in the presence of oriented trees. First, Subsection 2.4.1 looks at subdigraphs of tournaments with large χ, that is, of large order. Asymptotically, these are exactly acyclic digraphs. This directs our efforts towards obtaining tight bounds for oriented trees, paving the way towards questions for all digraphs. Then, Subsection 2.4.2 exhibits tight bounds for oriented trees in acyclic digraphs with high χ, which is an important subroutine for our later results.

2.4.1 Subdigraphs in tournaments: Sumner's conjecture

Restricting the problem to tournaments circumvents the obstacle of high girth and high chromatic number of Theorem 2.3.1, which allows us to ask for orientations of cycles. Still, transitive tournaments show that we cannot hope to force any directed cycles from high χ. In fact, we may always find a copy of TT_k in large enough tournaments, as shown by the following folklore lemma.

Lemma 2.4.1. *Any tournament of order 2^k contains the transitive tournament of order k.*

Proof. This clearly holds for $k = 1$. Let us assume the result holds up to k, and take any tournament T of order 2^{k+1}. Consider a vertex $v \in V(T)$ such that $d^+(v) \geq d^-(v)$, which exists by the handshaking lemma. Then, $N^+(v)$ has size at least 2^k, so it contain a copy of TT_k by induction, for which adding v yields TT_{k+1}. □

This means that asymptotically, the subdigraphs guaranteed by large enough tournaments are exactly the subdigraphs of TT_k.

Subtrees in tournaments

We should not expect this exponential dependency on the order to be required for all acyclic digraphs. The first interesting class in which (almost) tight bounds have been obtained are orientations of trees, which will be our concern in the general case. A precursory result on this front is Rédei's theorem, stating that any tournament admits a directed Hamiltonian path.

Theorem 2.4.2 (Rédei '34)**.** *Any tournament of order k contains a directed path of order k as a subgraph.*

Rosenfeld showed the above also holds when asking for antidirected paths [Ros72]. He further conjectured that apart from a few exceptions, any orientation of a Hamlitonian path exist, which was settled by Havet and Thomassé [HT00b]. Camion showed the existence of a directed Hamiltonian cycle [Cam59]. Again, Rosenfeld conjectured [Ros74] this to hold for every orientation, which was proven for $k \geq 68$ by Havet [Hav00].

Let us now consider general orientations of trees. There, observe $\chi \geq k$ is not sufficient for (oriented) trees of order k anymore, since diregular tournaments of order $2k-3$ forbid S_k^+ (the order k out-star). Still, it is easy to see that order $2k-2$ suffices for in and out stars (see Lemma 2.5.1). In 1971, Sumner conjectured this bound holds for all orientations of trees (see [Wor83]). Note that the following is exactly the restriction of Burr's conjecture to tournaments.

Conjecture 2.4.3 (Sumner '71)**.** *Any tournament of order* $2k-2$ *contains all oriented trees as subgraphs.*

One of the first classes of trees for which this was proven was out-arborescences, by a result of Havet and Thomassé [HT00a].

Theorem 2.4.4 (Havet, Thomassé '00)**.** *Any tournament of order* $2k-2$ *contains all out-arborescences of order* k *as a subgraph.*

Their proof was obtained by embedding the arborescences following a *median order* of the tournament. These are vertex orderings minimizing the number of backward arcs (which also witness a minimum feedback arc set). Using the same tool, the authors showed a bound of $(7k-5)/2$ for the general case, which was later improved by El Sahili to $3k-3$ in [El 04a]. The best bound so far is due to Havet and Dross [DH21].

Theorem 2.4.5 (Dross, Havet '21)**.** *Any tournament of order* $\frac{21}{8}(k-1)$ *contains all oriented trees on* k *vertices.*

Using tools from extremal graph theory, Kühn, Mycroft and Osthus achieved to prove Sumner's conjecture for large enough tournaments [KMO11].

Theorem 2.4.6 (Kuhn, Mycroft, Osthus '10)**.** *There exists an integer* k_0 *such that for any* $k \geq k_0$*, any tournament of order* $2k-2$ *contains all oriented trees of order* k *as subgraphs.*

The gap between the bounds of k and $2k-2$ for paths and trees suggests that Sumner's conjecture is not yet the end of the story. Considering stars are responsible for the $2k-2$ bound, Havet and Thomassé conjectured the following in [Hav03].

Conjecture 2.4.7 (Havet, Thomassé '03)**.** *Any tournament of order* $k+\ell-1$ *contains all oriented trees of order* k *with* ℓ *leaves.*

Towards this, Benford and Montgomery showed the existence of some universal C, such that tournaments of order $k+C\ell$ satisfy the conjecture [BM22].

2.4.2 Subtrees in bikernel-perfect digraphs and DAGs

Restricting our question on the universality of oriented trees within acyclic digraphs is justified because of a parallel to the "colour stacking" proof for trees of order k from $\chi \geq k$. In undirected graphs, a maximal independent set I is also a dominating set (corresponding to the first stacked class in Lemma 2.3.2). On the one hand, independence ensures that $\chi(G[V \setminus I]) \geq \chi(G) - 1$, on the other, domination allows us to glue a neighbour to any vertex of a possible tree in $G[V \setminus I]$. If we want to generalize this approach to any digraph D, we need (low χ) analogues of dominating sets that let us choose the direction of glued neighbours. An *in-dominating set* (respectively *out-dominating set*) of D is a subset $X \subseteq V$ such that every vertex outside of X is the in-neighbour (respectively out-neighbour) of a vertex in X. Now, an *in-kernel*, respectively an *out-kernel*, of D is an independent set that in-dominates D, respectively out-dominates D. Not all digraphs admit such kernels, but there always exists a *quasi-kernel*, an independent set Q at distance at most two [CL74] to all other vertices. Still, because we have no control over $\chi(N(Q))$, it seems difficult to derive oriented trees from quasi-kernels.

Even when digraphs do admit a kernel, their induced subgraphs may not, which becomes an issue later down in the induction. This motivates us to consider *bikernel-perfect* digraphs, those for which every subdigraph admits both an in-kernel and an out-kernel. The construction of oriented trees in bikernel-perfect digraphs, following the intuitions above, was shown Addario-Berry et al. [Add+13].

Theorem 2.4.8 (Addario-Berry et al. '13)**.** *Any bikernel-perfect digraph D such that $\chi(D) \geq k$ contains all oriented trees of order k.*

Proof. We show the result by induction on k, the case $k = 1$ being clear. Take a digraph D such that $\chi(D) \geq k + 1$, and take T to be any oriented tree of order $k + 1$. Without loss of generality, we may assume T has an out-leaf, say v, and let $T' = T - v$. Now, let K be an in-kernel of D, and let $D' = D - K$. Note that D' must have chromatic number at least k since K is independent, and by induction it contains a copy of T'. Then, the copy of the ancestor of v in T' has an out-arc towards K, yielding the desired copy of T. □

A notable class to which this applies are DAGs, as Richardson [Ric46] showed even digraphs forbidding odd directed cycles are bikernel-perfect.

Lemma 2.4.9 (Richardson '46)**.** *Acyclic digraphs are bikernel-perfect.*

Proof. We show the result by induction on the order n of the digraph, the result being clear when $n = 1$. Consider then an acyclic digraph D, of order $n > 1$, for which every subdigraph is bikernel-perfect. Take S to be the set of sinks in D, and observe that $V(D) \setminus N^-[S]$ contains no neighbour of S. Then, taking the union of S and an in-kernel for $D - N^-(S)$ yields an in-kernel of D. The case is symmetrical for out-kernels by taking sources of D. □

In Chapter 3, we show a (stronger) rooted variant of this due to [Add+13], and used throughout. The results above immediately yield the following corollary, which is tight by considering transitive tournaments.

Corollary 2.4.10. *Any acyclic digraph D such that $\chi(D) \geq k$ contains all oriented trees of order k.*

2.5 Subdigraphs from high chromatic number

In this section, we investigate the general case of oriented trees in digraphs with large chromatic number. Recall this question can only be asked for oriented trees because of Theorem 2.3.1. Following [Add+13], an oriented graph is *c-universal* if it is contained in every digraph with chromatic number at least c. There, tight bounds are known for out and in stars, as we will see in Subsection 2.5.1. Then, we look at oriented paths in Subsection 2.5.2 before tackling the question for oriented trees, which is Burr's conjecture, in Subsection 2.5.3.

2.5.1 Towards finding subdigraphs

In the case of undirected graphs, we have shown two techniques to obtain any tree of order k from $\chi \geq k-1$. We discuss how these fail to adapt to the directed setting, and see what can be recovered still. While we managed to adapt the "colour stacking" strategy to DAGs, this fails when applied to an arbitrary digraph D, as we cannot guarantee a (maximal) independent set with arcs in both directions towards the rest of the graph. The other strategy, extracting a subgraph of high minimum degree cannot be adapted to obtain both high minimum out *and* in degrees, which would be required to apply the same gluing strategy as Lemma 2.2.1. Still, high χ allows us to find (unrelated) vertices with *either* high in-degree, or high out-degree. The following bound coincides with the one for tournaments, and is thus tight.

Lemma 2.5.1. *Any digraph D such that $\chi(D) \geq 2k-2$ contains both the out-star S_k^+ and the in-star S_k^- of order k. That is, S_k^+ and S_k^- are $2k-2$-universal.*

Proof. Indeed, consider a digraph D that does not contain S_k^+, meaning $\Delta^+(D) \leq k-2$. Observe that for any subgraph H of D, the following holds:

$$2m(H) = \sum_{v \in V(H)} d_H(v) = \sum_{v \in V(H)} d_H^+(v) + d_H^-(v) = 2 \sum_{v \in V(H)} d_H^+(v) \leq 2(k-2)n(H)$$

Then, $\mathrm{ad}(H) = 2\frac{m(H)}{n(H)} \leq 2(k-2)$. In turn $\mathrm{mad}(D) \leq 2k-4$ and thus $\chi(D) \leq 2k-3$, which yields a contradiction. □

2.5.2 Oriented subpaths

The first result on the existence of a directed structure arising from chromatic number but not through degree conditions is the case of directed paths. This early result, which is a cornerstone in digraph structure, was obtained independently by Gallai [Gal68], Hasse [Has65], Roy [Roy67] and Vitaver [Vit62].

Theorem 2.5.2 (Gallai-Roy-Hasse-Vitaver Theorem '62)**.** *The directed path $\overrightarrow{P_k}$ is k-universal.*

For other orientations of P_k, Burr [Bur80] conjectured the same bound should hold (see also Chung [Chu81]).

Conjecture 2.5.3 (Burr '80)**.** *Every orientation of a path on k vertices is k-universal.*

We will see a quadratic bound in the chromatic number is sufficient, but no subquadratic bound had been achieved until now. Sitting in between $\overrightarrow{P_k}$ and general orientations are paths with b *blocks*, defined as those partitioning into k (maximal) directed paths. In light of the lack of progress towards Conjecture 2.5.3, El Sahili restricted Burr's question to 2-block paths [El 04b]. El Sahili and Kouider obtained a bound of $k+1$ in [SK07], and the conjecture was settled by Addario-Berry et al. [AHT07].

Theorem 2.5.4 (Addario-Berry et al. '07)**.** *Every oriented k-path with 2 blocks is k-universal.*

Directed and two-block paths are the only orientations for which Conjecture 2.5.3 is known in full. Still, we may relax the question and ask for linear bounds when the number of blocks is bounded.

Question 2.5.5. *For every $b \geq 3$, is there some c_b such that oriented k-paths with b blocks are $c_b k$-universal?*

A positive answer was obtained for 3-block paths by El Joubbeh [El 22], and for 4-block paths by El Joubbeh and Ghazal [EG24]. In Chapter 3, we give a positive answer to the question above, by showing the following bound, as well as settling the conjecture for 3-block paths.

Theorem 2.5.6. *For $b \geq 2$, every oriented k-path with b blocks is $\frac{(b+3)}{2}k$-universal.*

2.5.3 Oriented subtrees and Burr's conjecture

In this subsection, we look at obtaining linear bounds for any orientation of a tree. In the undirected case, a linear bound on χ in terms of k is sufficient to force all trees of order k. For digraphs, we know linear bounds for some orientations of paths, and Sumner's conjecture asks for a bound of $2k-2$ within tournaments. Could Sumner's conjecture then be lifted from tournaments to general digraphs? This question is exactly Burr's conjecture [Bur80], from the same paper as Conjecture 2.5.3.

Conjecture 2.5.7 (Burr '80)**.** *Every oriented tree of order k is $(2k-2)$-universal.*

Burr showed that trees of order k are $(k-1)^2$-universal, and this stood as the only bound for over 30 years. A decade ago, a factor 2 improvement was obtained by Addario-Berry et al. [Add+13]. Their paper shows quadratic bounds improving on Burr's through different approaches, broadly analogous the colour stacking and (iterated) minimum degree strategies for undirected graphs. In the following subsections, we present these two essential techniques, along with our path gluing lemma, which will be combined in Chapter 3 to yield our subquadratic bound.

Bound through maximal DAGs

With Corollary 2.4.10 in hand, note that if D contains an acyclic subdigraph with $\chi \geq k$ we can build any tree of order k. What happens otherwise, when all acyclic subdigraphs of D have $\chi < k$? If that is the case, Addario-Berry et al. [Add+13] have shown we can apply an analogue of the colour stacking technique (Lemma 2.3.2) for DAGs. This lead them to the following $\frac{k^2}{2}$ bound, which was the best bound for Burr's conjecture until now.

Theorem 2.5.8 (Addario-Berry et al. '13)**.** *Every oriented tree of order k is $(\frac{k^2}{2}-\frac{k}{2}+1)$-universal.*

Proof. We show the result by induction on k, letting $f(k) = (\frac{k^2}{2} - \frac{k}{2} + 1)$, the case $k = 1$ being clear. Assume the result holds up to a given k, let T be a tree of order $k+1$, take v to be any leaf of T, and let $T' = T - v$. Now, let D be a digraph with $\chi(D) \geq \frac{(k+1)^2}{2} - \frac{k+1}{2} + 1$, that is, $\chi(D) \geq f(k) + k$. Let $Y \subseteq V$ induce a maximal DAG of D, and note that if $\chi(Y) \geq k+1$, then Corollary 2.4.10 yields the result. Therefore, $\chi(D[Y]) \leq k$, so $\chi(D-Y) \geq f(k)$, meaning we may find an induced copy of T' in $D - Y$. Then, as Y is a maximal DAG, the copy of the ancestor of v has both in and out neighbours in Y, yielding a copy of T. $\square$

A slightly different view of this is to see that if $\chi < k$ for every acyclic subgraph of D, then the dichromatic number must be (relatively) large. Indeed, if D has chromatic number $k(k-1)$, then we have $\overrightarrow{\chi}(D) \geq k$. Below, we show an analogue of Lemma 2.3.2 for $\overrightarrow{\chi}$, stacking maximal

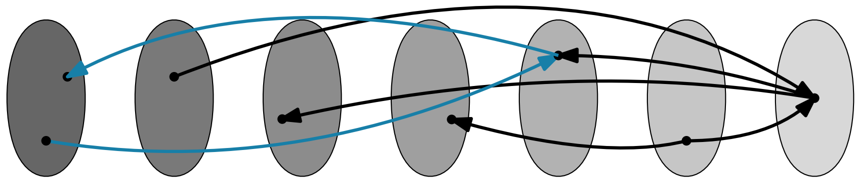


Figure 2.2: A dicolouring obtained by stacking classes as successive maximal DAGs, from the left. Any oriented tree built on the last 6 colour classes may be extended from any vertex in both directions towards the first class, yielding all trees of order 7.

DAGs, which yields a bound of $k(k-1)$ for Burr's conjecture. An illustration reminiscent of the undirected case is shown in Figure 2.2.

Lemma 2.5.9. *If $\vec{\chi}(D) \geq k$, then D contains all oriented trees of order k.*

Proof. The result clearly holds for $k = 1$. Then, assume it holds for some k and consider D with $\vec{\chi}(D) \geq k+1$ as well as an oriented tree T of order k. Let $Y \subseteq V$ induce a maximal DAG in D, and define $D' = D - Y$, noting $\vec{\chi}(D') \geq k$. By induction, let T' be a copy in D' of some subtree of T obtained by removing a single leaf. Then, since Y is maximal, it is both in-dominating and out-dominating in D. This lets us consider the parent of the removed leaf, and append the leaf back with any orientation, giving us T. $\square$

In fact, high $\vec{\chi}$ is a powerful condition, and the question of which (induced) subgraphs it implies will be the subject of Section 2.6.

Bound by gluing leaves

We present the strategy used in Burr's $(k-1)^2+1$ bound from [Bur80], in the language of [Add+13]. In the following, for any oriented tree T, let $Out(T)$ be the set of its out-leaves, and $In(T)$ be the set of its in-leaves. The operation of "gluing" leaves used by Burr derives a universality bound for T from a bound on $T - Out(T)$, as follows.

Lemma 2.5.10 (Burr '80)**.** *If T is an oriented tree of order k, and $T - Out(T)$ (respectively $T - In(T)$) is c-universal, then T is $c + 2k - 2$-universal.*

Proof. Take T to be a tree of order k, and assume $T - Out(T)$ is c-universal. Consider D to be any digraph such that $\chi(D) \geq c + 2k - 2$. We partition $V(D)$ into two sets, letting Z be the subgraph induced by the vertices of D for which $d^+ < k - 1$, and $X = V(D) \setminus Z$. Then, by the handshaking lemma, $\text{mad}(D[Z]) \leq 2(k-2)$, so $\chi(D[Z]) \leq 2k - 3$. This means $\chi(D[X]) \geq c + 1$, which allows us to find a copy T' of $T - Out(T)$ in X. Then, all vertices of this copy have out-degree at least $k - 1$, and as long as $|V(T')| \leq k - 1$, we can successively append out-leaves distinct from T' to any vertex of T', which allows us to build a copy of T, see Figure 2.3. The proof is symmetrical for in-leaves, by taking Z to be the subgraph induced by vertices v for which $d^-(v) < k - 1$ instead. $\square$

Gluing leaves successively through the above, Burr's obtained the following quadratic bound [Bur80].

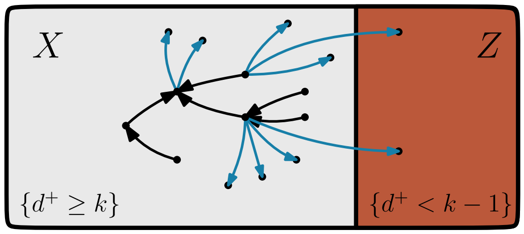


Figure 2.3: Gluing leaves to obtain T from $T - Out(T)$, deriving a universality bound for T. The copy of $T - Out(T)$ is shown in black, and belongs to a subgraph X of high enough out-degree to ensure all out-leaves (in blue) may be glued.

Corollary 2.5.11 (Burr '80)**.** *Every oriented tree of order k is $k^2 - k$-universal.*

Proof. It is clear that the bound holds for $k \leq 2$, letting us assume the result for some $k \geq 2$ and consider an oriented tree T of order $k+1$. If T contains an out-leaf, let $T' = T - Out(T)$, else T contains an in-leaf and we let $T' = T - In(T)$. Now, $|V(T')| \leq k$, so T' is $k^2 - k$-universal by induction, and Lemma 2.5.10 yields that T is $k^2 - k + (2(k+1) - 2)$-universal, thus $(k+1)^2 - (k+1)$-universal. □

The gluing operation in the proof of Lemma 2.5.10 was later refined in [Add+13] to obtain an increase of $2k - 4$. When deriving a bound through repeated application of this, the best outcomes come for oriented trees with "low depth", requiring few gluing of in-leaves or out-leaves to retrieve the whole tree. In [Add+13], the authors define $\mathrm{st}(T)$ to be the minimum number of such operations, and obtain a $O(k \cdot \mathrm{st}(T))$ bound. Using this, they obtain a bound of $k^2 - 3k + 4$ for all oriented trees, improving on Burr's bound, but still short of the $\frac{1}{2}$ factor in the k^2 term of Theorem 2.5.8. Still, the directed path $\overrightarrow{P_k}$ requires $k - 1$ gluing operations, which attains the quadratic bound. On the other hand, we know by Theorem 2.5.2 that directed paths only require chromatic number k, and this is even true for paths with 2 blocks.

Towards gluing paths

In order to improve on the quadratic bound, it is then natural to ask for an operation gluing directed paths.

Question 2.5.12. *Is there some linear $f(k)$ such that, if T is c-universal, then the tree obtained from T by gluing a directed path of length k is $c + f(k)$-universal?*

Our main contribution in Chapter 3 is a positive answer to this question, obtained by deriving a lemma to glue directed paths. This directly yields our linear bound for b-block paths Theorem 2.5.6 when b is fixed. Then, the combination of paths gluing and leaves gluing operations (according to the number of leaves) allows us to derive a subquadratic bound for arborescences. We show a corresponding gluing lemma for oriented paths, where $f(k)$ is quadratic, but a simple tweak on the induction allows us to show our general subquadratic bound for Conjecture 2.5.7.

Theorem 2.5.13 (Bessy, R, Gonçalves '25)**.** *Every oriented tree of order k is* $(8\sqrt{\frac{2}{15}}k\sqrt{k}+\frac{11}{3}k+\sqrt{\frac{5}{6}}\sqrt{k}+1)$*-universal.*

2.5.4 Around Burr's conjecture

In the spirit of Conjecture 2.4.7 for tournaments, Havet conjectured the following bound on the universality of trees parameterized by number of leaves (see [DH21]).

Conjecture 2.5.14. *Every oriented tree of order k with ℓ leaves is $k+\ell-1$-universal.*

Considering the lack of even linear bounds for Burr's conjecture, the most pressing open question supporting the above would be Burr's path conjecture (2.5.3).

A more promising direction for the time being is to look for weakenings of the conjecture. In his thesis, Lochet [Loc18] proposed a subdivision variant of the conjecture by asking the following.

Conjecture 2.5.15 (Lochet '18)**.** *There exists some C such that every digraph D with $\chi(D) \geq Ck$ contains every oriented tree of order k as a subdivision.*

Note that the weakening of Conjecture 2.5.15 asking for immersions instead of subdivisions is also of interest.

Towards improving on our subquadratic bound, an interesting direction, first proposed by Addario-Berry et al. [Add+13], is to look for in-(out-)dominating sets with small χ. This would generalize Theorem 2.4.8, for DAGs, to all digraphs. Already for tournaments, an easy inductive argument yields the following.

Observation 2.5.16. *Any tournament of order 2^k admits in-dominating and out-dominating sets of size at most k.*

To our knowledge, the analogue question for digraphs has not been raised, motivating the following generalization.

Conjecture 2.5.17 (Bessy, R, Gonçalves '25)**.** *Every digraph D admits in-dominating and out-dominating sets with chromatic number at most* $\lceil \log(\chi(D)) \rceil$

This conjecture and the observation above would be tight, as Graham and Spencer have shown Paley tournaments match this upper bound [GS71]. A direct corollary of the above, using the same strategy as Theorem 2.4.8, would yield a $k\lceil \log(k) \rceil$ bound for Burr's conjecture.

Oriented cycles

As in the undirected case, we can only ask for orientations of cycles in terms of subdivisions. In all generality, this is not possible for any cycle because of acyclic digraphs, as shown by Cohen et al. [Coh+18b].

Theorem 2.5.18 (Cohen et al. '18)**.** *For any c, b, there exists an acyclic digraph with chromatic number c in which all oriented cycles have at least b blocks.*

Nevertheless, the question becomes interesting again when considering strongly connected digraphs, thanks to the following result of Bondy [Bon76].

Theorem 2.5.19 (Bondy '76). *Every strongly-connected digraph D such that $\chi(D) \geq k$ contains a subdivision of $\overrightarrow{C_k}$.*

Cohen et al. conjectured that any subdivision of oriented cycles may be obtained asymptotically [Coh+18b].

Conjecture 2.5.20 (Cohen et al. '18). *For every oriented cycle C, there exists $f(C)$ such that any strongly-connected digraph D such that $\chi(D) \geq f(C)$ contains a subdivision of C.*

They showed the above for 2-block cycles [Coh+18a], and Kim et al. derived an upper bound on $f(C)$ that is quadratic in $|C|$ [Kim+18].

Induced subtrees

For subgraphs, it is already striking that an undirected parameter such as χ is able to force directed substructures with roughly the same budget as it does undirected ones. How strong is this behaviour? Is it possible to ask for *induced* oriented trees from high χ, given some sparsity condition? Going back to asymptotic considerations, this is asking for a directed analogue of the Gyárfás-Sumner conjecture (1.1.19). The most audacious way to do so is by saying an oriented tree T is χ-bounding if there exists a function f such that induced-T-free digraphs with $\chi \geq f(k)$ contain a clique of order k. Then, we ask whether all oriented trees are χ-bounding. This generalization was first considered by Kierstead and Trotter [KT92], which disproved it through the following.

Theorem 2.5.21 (Kierstead, Trotter '92). *There exist digraphs of arbitrarily large chromatic number with no triangles nor an induced copy of $\overrightarrow{P_4}$.*

Their construction builds upon Zykov's triangle-free graphs with large χ [Zyk49]. Another negative answer had already been given by Gyárfás for the alternating (antidirected) path of length 4, by forbidding it within a certain orientation of shift graphs [Gyá89].

On the other hand, Chudnovsky et al. [CSS19] gave a positive answer for the two other orientations of P_4 (up to symmetry), as well as for oriented stars.

Theorem 2.5.22 (Chudnovsky et al. '19). *The following oriented trees are χ-bounding:*

- *Any orientation of a star,*
- $\overrightarrow{Q_4} = \rightarrow\leftarrow\leftarrow$
- $\overrightarrow{Q_4}' = \leftarrow\leftarrow\rightarrow$

The positive and negative answers above characterize all χ-binding oriented trees of order at most 4. Yet there remain orientations of 5-vertex trees for which nothing is known, prompting the following question.

Question 2.5.23. *Give a full characterization of all χ-binding oriented trees.*

In the following section, we bring hopes of a directed Gyárfás-Sumner conjecture back with dichromatic number.

2.6 Subdigraphs from high dichromatic number

In this section, we are concerned with obtaining (induced) subdigraphs from large dichromatic number. We investigate the question for (non-induced) subdigraphs and subdivisions in Subsection 2.6.1. Then, we consider the induced subdigraphs of tournaments in Subsection 2.6.2, and those of dense classes Subsection 2.6.3. Then, Subsection 2.6.1 looks at the general case, with much less history, and we lay there the ground for our contributions in Chapter 4.

2.6.1 Subdigraphs and subdivisions

Note that if $\overrightarrow{\chi} \geq k$, then $\chi \geq k$ as well as $\delta^+, \delta^- \geq k-1$, so all previous results about subdigraphs from high χ or high degrees apply. In Subsection 2.2.2, we mentioned an obstacle to cycles for degree variants. This is in fact an obstacle for $\overrightarrow{\chi}$ in the first place, as shown by Harutyunyan and Mohar [HM12].

Theorem 2.6.1 (Harutyunyan, Mohar '12)**.** *There exists digraphs with arbitrarily large (undirected) girth and dichromatic number.*

This completely settles the question: only oriented trees may be obtained, and the bound of $\overrightarrow{\chi} \geq k$ for trees of order k is tight.

If we ask for more than trees contained as subdivisions, Mader's Conjecture 2.2.10 for out-degree would imply any TT_k for high $\overrightarrow{\chi}$. Still, dichromatic number proves to be significantly stronger, as witnessed by the following result by Aboulker et al. [Abo+19].

Theorem 2.6.2 (Aboulker et al. '19)**.** *For any digraph H on n vertices and m arcs, any digraph D such that $\overrightarrow{\chi}(D) \geq 4^m(n-1)+1$ contains a subdivision of H.*

2.6.2 Subtournaments in tournaments

The first results dealing with *induced* subdigraphs from high $\overrightarrow{\chi}$ were obtained by Berger et al. [Ber+13], by considering subtournaments of high $\overrightarrow{\chi}$ tournaments. There, we already know large $\overrightarrow{\chi}$ yields all acyclic digraphs asymptotically (because high χ does). If we want to ask for more, observe that $\overrightarrow{\chi} \geq 2$ already guarantees $\overrightarrow{C_3}$ as a subgraph. Let us say a tournament H is a *hero* if the class of tournaments forbidding H has bounded dichromatic number. Equivalently, these are the tournaments forced by sufficiently large $\overrightarrow{\chi}$, within the class of tournaments. The authors of [Ber+13] fully characterize heroes in tournaments by an inductive description.

2.6.3 Subdigraphs in dense digraphs

Before considering general digraphs, it is natural to investigate our question for a class sparser than tournaments, but still dense in some sense. Aboulker, Charbit and Naserasr asked for the subgraphs (superheroes) forced by high $\overrightarrow{\chi}$ in classes of bounded independence number α. Of course, these subgraphs should include all heroes, as $\alpha = 1$ for tournaments, but we may expect more. This question was completely settled by Harutyunyan, Le, Newman and Thomassé [Har+17], who showed that we cannot ask for more: the subdigraphs of high $\overrightarrow{\chi}$ and bounded α graphs are exactly the heroes of [Ber+13].

2.6.4 Induced subdigraphs: extending the Gyárfás-Sumner conjecture

Adapting the (undirected) ideas of [CS14], a set $\mathcal{F}$ of digraphs is called *heroic* if the class of (induced) $\mathcal{F}$-free digraphs has bounded dichromatic number. Now, every heroic set $\mathcal{F}$ must contain a digraph containing a digon, else $\mathcal{F}$-free digraphs would contain all bidirected graphs. Moreover, $\mathcal{F}$ should also contain some oriented graph, otherwise all oriented graphs would be allowed. We are far from a characterization such as the one in [Ber+13] for all digraphs, but partial results have been obtained for more general subclasses, see [AAC24, AAS22, Car+25].

Going back to the root of these questions, we should ask for a directed variant of the Gyárfás-Sumner conjecture. Recall that this fails for χ because of directed and alternating orientations of P_4 (see Theorem 2.5.21). Aboulker, Charbit and Naserasr initiated the study of directed analogues using $\vec{\chi}$ in [ACN21]. For a purely directed analogue, they consider only oriented graphs, which translates as looking for heroic sets $\mathcal{F}$ containing $\overleftrightarrow{K_2}$. It is easy to see that the only heroic set of size 2 is $\mathcal{F} = \{\overleftrightarrow{K_2}, \overrightarrow{K_2}\}$, so we are interested in heroic sets $\mathcal{F}$ of size three. Our set $\mathcal{F}$ should also include some fixed hero, which we call H, for otherwise $\mathcal{F}$-free graphs would contain tournaments with arbitrarily large $\vec{\chi}$. For the third member, note that Theorem 2.6.1 forces us to choose an oriented forest F. In [ACN21], the authors show that if such a set $\{\overleftrightarrow{K_2}, H, F\}$ is heroic, then either F is the disjoint union of oriented stars, or H is a transitive tournament. They conjectured the converse holds, but taking F to be the union of K_1 and $\overrightarrow{P_2}$, Aboulker et al. [AAC24] disproved that all heroes H bound $\vec{\chi}$.

Therefore, we are interested in the surviving part of the conjecture, when H is a transitive tournament, and F can be any oriented forest. Recall that forbidding TT_k also forbids any tournament of order more than 2^k, which motivates the following analogue of χ-boundedness. We define a class of digraphs $\mathcal{D}$ as $\vec{\chi}$*-bounded* if there exists a function f such that every $D \in \mathcal{D}$ satisfies $\vec{\chi}(D) \leq f(\omega(D))$. We say a digraph H is $\vec{\chi}$*-bounding* if the class of H-free oriented graphs is $\vec{\chi}$-bounded. We are now ready to state the surviving part of the conjecture in terms of $\vec{\chi}$-boundedness.

Conjecture 2.6.3 (Aboulker, Charbit, Naserasr '21)**.** *Every oriented forest is* $\vec{\chi}$*-bounding.*

Every χ-bounding digraph is $\vec{\chi}$-bounding, meaning the conjecture holds for oriented stars, as well as $\rightarrow\leftarrow\leftarrow$ and $\leftarrow\leftarrow\rightarrow$ (see 2.5.4). In support of Conjecture 2.6.3, the authors showed the case $k = 4$ and $F = \overrightarrow{P_4}$, through their definition of a tool called *dipolar sets*. A result of Steiner [Ste23] also shows the conjecture for $k = 3$ and $F = \overrightarrow{A_4}$, the alternating (antidirected) orientation of P_4. The first open cases for general k are the two orientations of P_4 that are exactly the non χ-bounding 4-vertex oriented trees. Our main contribution in Chapter 4, crucially reliant on dipolar sets, is a proof of Conjecture 2.6.3 for $\overrightarrow{P_4}$ and $\overrightarrow{A_4}$, achieving to settle the question for orientations of P_4.

Theorem 2.6.4 (Cook, Masařík, Pilipczuk, R., Souza '20)**.** *All orientations of* P_4 *are* $\vec{\chi}$*-bounding.*

We note that following our result, Aboulker et al. [Abo+24] showed the result for $\overrightarrow{P_6}$ and $k = 3$. The conjecture is still widely open, and it is not known in general for any oriented tree on order at least 5 other than stars. In particular, we lack an analogue of the famous Gyárfás path argument (Theorem 2.3.4), and the case of $\overrightarrow{P_n}$ is still open.

Infinite heroic sets

For any finite list of digraphs $D_1, D_2, \ldots, D_k$, if the class of $(D_1, D_2, \ldots, D_k)$-free oriented graphs is $\vec{\chi}$-bounded, at least one D_i is an oriented forest. Indeed, otherwise recalling Theorem 2.6.1, digraphs with sufficiently large girth and arbitrary $\vec{\chi}$ would forbid $(D_i)_i$. One might ask whether the situation changes when the list of forbidden digraphs is infinite. We list some results related to this:

- In [Car+23], Carbonero, Hompe, Moore, and Spirkl provided a construction for oriented graphs with clique number at most three, arbitrarily high dichromatic number, and no induced directed cycles of odd length at least 5. They use this to show that there exist graphs G with arbitrarily large chromatic number such that every induced triangle-free subgraph of G has chromatic number at most four, disproving a well-known conjecture, attributed to Esperet in [SS20a].
- In [ABV22], Aboulker, Bousquet, and de Verclos showed that the class of chordal oriented graphs, that is, oriented graphs forbidding induced directed cycles of length greater than three, is not $\vec{\chi}$-bounded, answering a question posed in [Car+23].
- In [Car+22], Carbonero, Hompe, Moore, and Spirkl extended the result of [Car+23] to t-chordal graphs. A digraph is *t-chordal* if it does not contain an induced directed cycle of length other than t. In [Car+22] the authors showed that t-chordal graphs are not $\vec{\chi}$-bounded, but t-chordal $\vec{P_t}$-free graphs are $\vec{\chi}$-bounded.

2.6.5 More directed analogues of Gyárfás-Sumner

Considering the apparent hardness of Conjecture 2.6.3, it is natural to relax the question by asking whether forbidding a forest in the underlying graph, or equivalently, all orientations of a given forest, yields a $\vec{\chi}$-bounded class. Let us define the dichromatic number $\vec{\chi}(G)$ of an undirected graph G as the maximal dichromatic number over all its orientations. Because $\vec{\chi}(G) \leq \chi(G)$, the question above also weakens the Gyárfás-Sumner conjecture to the following, which was asked by Aboulker at various workshops [Abo25].

Conjecture 2.6.5 (Aboulker)**.** *For any tree T and any integer k, there exists some $c_{T,k}$ such that any graph G with $\vec{\chi}(G) \geq c_{T,k}$ contains either an induced T, or K_k.*

Let us now recall an intriguing conjecture of Erdős and Neumann-Lara [Erd79b] below.

Conjecture 2.6.6 (Erdős, Neumann-Lara '79)**.** *There exists a function f such that if a graph G satisfies $\vec{\chi}(G) \leq k$ then, $\chi(G) \leq f(k)$.*

With our definitions above, this would yield that for any graph G, $\vec{\chi}(G) \leq \chi(G) \leq f(\vec{\chi}(G))$. In turn, we remark that proving Conjecture 2.6.6 would yield an equivalence between Conjecture 2.6.5 and the Gyárfás-Sumner conjecture.

On a different note, another variant of the Gyárfás-Sumner conjecture was recently proposed for tournaments by Aboulker et al. in [Abo+23]. Their definitions rely on “backedge” graphs of tournaments, which are (undirected) graphs obtained by ordering $V(T)$ and considering only edges going backwards. The motivating observation behind considering this is that $\vec{\chi}(T)$ is equal to the minimum of χ over all backedge graphs of T. The authors define the clique number $\vec{\omega}(T)$ of T as the minimum clique number of its backedge graphs, and conjecture the following, after proving the converse direction.

Conjecture 2.6.7 (Aboulker et al. '23)**.** *For any tournament H admitting a backedge graph that is a forest, and any integer k, there exists some $c_{k,H}$ such that every tournament T with $\vec{\chi}(T) \geq c_{k,H}$ either contains H, or satisfies $\omega(T) \geq k$.*

2.7 When χ exceeds degrees: Brooks analogues

This section treats analogues of Brooks's theorem, introducing the problematics towards Chapter 5. In the previous sections, we asked for substructures guaranteed across all (di)graphs whenever χ or $\vec{\chi}$ exceeds some fixed value. Here, we aim at understanding the structure of (di)graphs with high chromatic number, or variants of it, in a much stronger sense: when it exceeds their maximal degree. Such a requirement typically not only forces certain subgraphs, but ensures the whole graph is one of a few "obstructions".

For undirected graphs and usual colouring, our question is answered exactly by Brooks's theorem, which (slightly restated) says that $\chi > \Delta$ implies the whole graph is either a clique or an odd cycle. Our goal here will be to obtain analogues of this statement for more general colourings. From now on, we frame this as looking for generalizations of Brooks's theorem, as answers to the following (very) broad question.

Question 2.7.1. *What colouring budget, in terms of degree(s), ensures "colourability" except for a finite set of characterizable obstructions?*

Brooks's analogues were obtained as answers to the above for various colouring notions, and this is an active line of research, especially following the introduction of new colouring variants. The proofs of these analogues usually imply the existence of a polynomial-time algorithm to either produce the required colouring, or decide it does not exist. Nevertheless, linear-time algorithms are known only for a few cases. Throughout this section, linear means $O(m)$, as the input size is the size of teh graph.

The very nature of our question, asking for an exact characterization of obstructions to colouring with an "almost maximal" budget, does not lend itself to incremental progress. Indeed, Brooks's analogues have been historically proved in full, and research progresses rather by showing analogues for increasingly general notions. We distinguish different generalizing directions, for each, we introduce both the undirected and directed Brooks's analogues, and associated algorithms. We begin in Subsection 2.7.1 by recalling Brooks's theorem for χ and Δ, as well as its (directed) analogue for $\vec{\chi}$ and Δ_{max}, digressing on extensions orthogonal to Question 2.7.1. In Subsection 2.7.2, we investigate a first type of generalization, studying analogues for list colourings and dicolourings. Then, a different direction is explored in Subsection 2.7.3 with analogues for the colouring variant given by degeneracy partitions. We unite these two analogues in Subsection 2.7.4 through variable degeneracy partitions. Presenting directed generalizations of the latter is the object of Chapter 5, where we introduce "bivariable" degeneracy partitions.

2.7.1 Brooks's theorem for (di)colourings

Let us start by recalling Brooks's theorem [Bro41], which stands as one of the most fundamental results in structural graph theory. For a comprehensive literature, we refer the interested reader to the recent book on the topic by Stiebitz, Schweser and Toft [SST24].

Theorem 2.7.2 (Brooks '41)**.** *If G is connected, then $\chi(G) \leq \Delta(G) + 1$, and equality holds if and only if G is an odd cycle or a clique.*

Algorithmically, deciding between chromatic number Δ and $\Delta + 1$ is one of the few restrictions of k-colourability that is solvable in polynomial-time. Indeed, the original proof of Brooks's theorem can be adapted into a quadratic-time algorithm to construct such a Δ-colouring when it exists. Later, Lovász provided a simpler proof [Lov75], which leads to a linear-time algorithm, as shown by Baetz and Wood [BW14]. An alternative linear-time algorithm was also given by Skulrattanakulchai [Sku06]. If we ask the (less restrictive) question of producing a $(\Delta + 1)$-colouring, this can be done in linear time. Moreover, Assadi et al. have recently broken the linearity barrier for several models of sublinear algorithms [ACK19].

In the directed case, a similar behaviour is exhibited by dichromatic number with respect to Δ_{max}. We know $\chi \leq \Delta_{max} + 1$, and let us recall the directed analogue obtained by Harutyunyan and Mohar [HM11].

Theorem 2.7.3 (Harutyunyan and Mohar '12)**.** *Let $D = (V, A)$ be a connected digraph. Then*

$$\vec{\chi}(D) \leq \Delta_{max}(D) + 1$$

and equality holds if and only if D is a directed cycle, a bidirected odd cycle, or a biclique.

The constructive proof of the result yields a linear-time algorithm, as shown in [HM11]. In the special case of oriented graphs, the strengthening of Theorem 2.7.3 for $\vec{\chi}(D)$ and Δ_{min} of Picasarri-Arrieta [Pic24b] admits a linear-time algorithm for finding a dicolouring of D using at most Δ_{min} if it exists.

More than degrees

Apart from degrees, Brooks's analogues have also been obtained by generalizing constraints on Δ to connectivity conditions. For example, Aboulker et al. [Abo+17] studied an edge-connectivity variant, eventually settled by Stiebitz and Toft [ST18]. The generalization to arc-connectivity was obtained Aboulker et al. [AAC23].

Another interesting conjecture, proposed by Reed [Ree98], sits midway between the $\Delta+1$ bound for χ and χ-boundedness.

Conjecture 2.7.4 (Reed '98)**.** *For any graph G, $\chi(G) \leq \lceil(\Delta(G) + \omega(G) + 1)/2\rceil$.*

Reed first showed the existence of an $\epsilon > 0$ such that a $\chi \leq (1-\epsilon)(\Delta+1)+\epsilon\omega$, but the conjecture stands to this day. The conjecture was generalized to dichromatic number by Kawarabayashi and Picasarri-Arrieta [KP25], who proved an analogue to the above.

2.7.2 List-(di)colouring analogues

One of the first colouring variants considered for a generalization of Brooks's theorem are list colourings. We present the existing analogues of Brooks's in the case of list colourings and dicolourings, and we will see obstructions to colourability are now formed by "gluing" obstructions presented in the last subsection.

Given a graph G, a *list assignment* L is a function that associates a list of "available" colours to every vertex of G. An *L-colouring* of G is then a proper colouring α of G such that $\alpha(v) \in L(v)$ for

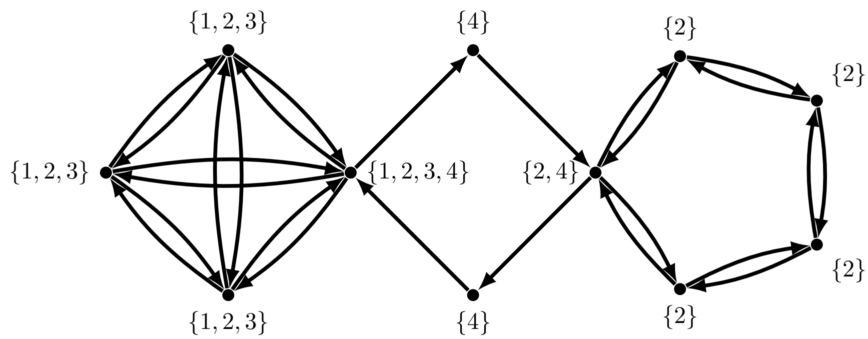


Figure 2.4: A directed Gallai tree, with a list assignment L satisfying $|L(v)| \geq \max(d^+(v), d^-(v))$ at each vertex v, and yielding an obstruction to Theorem 2.7.6.

every vertex v. Instantiating Question 2.7.1, the natural extension of Brooks's theorem here is to ask when $|L(v)| \geq d(v)$ for all v is a sufficient condition for an L-colouring. Such a characterization was given independently in 1979 by Borodin [Bor79], and Erdős, Rubin and Taylor [Erd79a]. The graphs for which this condition is not sufficient are *Gallai trees*, which are connected graphs in which every maximal biconnected subgraph, or *block*, is either a complete graph or an odd cycle.

Theorem 2.7.5 (Borodin [Bor79] ; Erdős, Rubin, Taylor [Erd79a]). *Let $G = (V, E)$ be a connected graph and L be a list assignment of G such that, for every vertex $v \in V$, $|L(v)| \geq d(v)$. If G is not L-colourable, then G is a Gallai tree and $|L(v)| = d(v)$ for every vertex v.*

This result actually generalizes Brooks's theorem by setting $L(v) = [\Delta]$ for every $v \in V(G)$, and noting a minimum counter-example cannot admit a 1-separator. The procedures given in the proofs leads to polynomial time algorithms, but the first linear-time algorithm finding such a colouring was given by Skulrattanakulchai [Sku06].

In the directed case, Theorem 2.7.3 from [HM11] was in fact already shown for list dicolourings. The obstructions in this case are analogue to the undirected version. A *directed Gallai tree* is a digraph in which every block is a directed cycle, a bidirected odd cycle or a bidirected complete graph, see Subsection 2.7.2. Given a list assignment of a digraph D, an L-dicolouring α is a dicolouring of D such that $\alpha(v) \in L(v)$ holds for every vertex v of D.

Theorem 2.7.6 (Harutyunyan and Mohar [HM11]). *Let $D = (V, A)$ be a connected digraph and L a list assignment of D such that $|L(v)| \geq \max(d^+(v), d^-(v))$ holds for every vertex $v \in V$. If D is not L-dicolourable, then D is a directed Gallai tree.*

This also generalizes Theorem 2.7.5, and it is shown in [HM11] that such a colouring may be found in linear $O(m)$ time when it exists.

2.7.3 Degeneracy partitions

Another generalization of colouring for which an analogue of Brooks's theorem has been obtained relaxes the independence of colour classes to a bounded degeneracy condition. This is sensible as independent sets are exactly 0-degenerate graphs. We present the analogue for undirected graphs, as the directed variant was only obtained from a stronger result presented in the next subsection.

Let $P = (p_1, \dots, p_s)$ be a sequence of positive integers. A graph G is *P-colourable* if there exists an s-colouring α of G such that, for every $i \in [s]$, the subgraph of G induced by the colour class $\alpha^{-1}(i)$ is $(p_i - 1)$-degenerate. When $p_1 = \dots = p_s = 1$, observe that a P-colouring is exactly a proper s-colouring. Borodin [Bor76], and Bollobás and Manvel [BM79] independently proved the following.

Theorem 2.7.7 (Borodin [Bor76] ; Bollobas and Manvel [BM79])**.** *Let G be a connected graph with maximum degree Δ and $P = (p_1, \dots, p_s)$ be a sequence of $s \geq 2$ positive integers such that $\sum_{i=1}^{s} p_i \geq \Delta$. If G is not P-colourable, then $\sum_{i=1}^{s} p_i = \Delta$ and G is a complete graph or an odd cycle.*

Brooks's theorem follows from Theorem 2.7.7 by setting $s = \Delta$ and $p_i = 1$ for every $i \in [\Delta]$. In terms of complexity, the proofs of [Bor76] and [BM79] provide at best cubic algorithms, which were recently improved to a linear-time algorithm by Corsini et al. [Cor+23].

2.7.4 Variable degeneracy

We have seen two orthogonal directions for generalizing Brooks's theorem: list colouring and degeneracy partitions. One can naturally expect an even more general theorem subsuming both Theorems 2.7.5 and Theorem 2.7.7. Such a result was obtained by Borodin, Kostochka, and Toft through the introduction of *variable degeneracy* [BKT00].

Let $G = (V, E)$ be a graph and $f : V \to \mathbb{N}$ be an integer-valued function. We say that G is *strictly-f-degenerate* if every subgraph H of G contains a vertex v satisfying $d_H(v) < f(v)$. Let $s \geq 1$ be an integer and $F = (f_1, \dots, f_s)$ be a sequence of integer-valued functions. The graph G is *F-colourable* if there exists an s-colouring α of G such that, for every $i \in [s]$, the subgraph of G induced by the vertices coloured i is strictly-f_i-degenerate. Then, the authors consider instances (G, F) where for every vertex $v \in V(G)$, $\sum_{i=1}^{s} f_i(v)$ is at least $d(v)$, which can be seen as allowing a "colour budget". They give an explicit recursive characterization of *hard pairs*, instances satisfying the above condition that are not F-colourable. Informally, these instances are a generalization of Gallai trees, where some "monochromatic" blocks are allowed, and in which the condition above is tight in every vertex. In [BKT00], the authors prove the following theorem, generalising both Theorems 2.7.5 and 2.7.7.

Theorem 2.7.8 (Borodin, et al. '00)**.** *Let G be a connected graph and $F = (f_1, \dots, f_s)$ be a sequence of integer-valued functions such that, for every vertex $v \in V(G)$, $\sum_{i=1}^{s} f_i(v) \geq d(v)$. Then G is F-colourable if and only if (G, F) is not a hard pair.*

Note that Theorem 2.7.7 for degeneracy partitions can be obtained from the result above by setting f_i to the constant function equal to p_i for every $i \in [s]$. This is also the case of list-colouring, where a list assignment L can be simulated by setting $f_i(v)$ to 1 when $i \in L(v)$, and 0 otherwise, generalizing Theorem 2.7.5. While these two previous specializations admit linear-time algorithms, this is not the case for Theorem 2.7.8.

Let us also mention that Theorem 2.7.8 has been extended to hypergraphs, see [SS21a, SS21b], as well as to correspondence colouring, see [KSS23].

Directed analogues of variable degeneracy

Our main concern in Chapter 5 is with generalizations of Theorem 2.7.8 to digraphs, which would yield all analogues presented so far as special cases. While degeneracy is a clear notion in the undirected case, there are multiple ways of defining directed degeneracy variants. The first generalization using a directed variable degeneracy notion was introduced by Bang-Jensen et al. [BSS22], and we give its full definition in Chapter 5. Here, our initial motivation is the fact that in both this analogue and Theorem 2.7.8, the algorithms producing colourings or certifying the obstructions are quadratic at best.

Our goal is to obtain a linear algorithm for these two variants, matching those for weaker analogues. Towards this, we define a variant of degeneracy for digraphs, dubbed *variable bidegeneracy*. Based on this notion, we introduce F-dicolourings, which are partitions into digraphs satisfying bivariable degeneracy constrains. This generalizes the colouring variant considered in [BSS22], and all others in turn. We answer Question 2.7.1 for our notion, with a corresponding Brooks analogue having the same obstructions as [BSS22]. Moreover, we provide a linear-time algorithm either outputting a partition, or certifying through a hard pair that none exist. Restricting this to previous notions yields linear-time algorithms in all cases, and in particular for (directed) variable degeneracy analogues [BSS22, BKT00].

Chapter 3

Oriented trees in $O(k\sqrt{k})$-chromatic digraphs

This chapter presents our subquadratic bound for Burr's conjecture, following the introduction given in Section 2.5. It is based on joint work with Stéphane Bessy and Daniel Gonçalves, appearing in [BGR24].

3.1 Introduction

Recall a digraph is *c-universal* for some value $c > 0$ if every digraph with chromatic number at least c contains it as a subdigraph. We have seen that this can only be expected of oriented trees, and recall Burr's conjecture [Bur80].

Conjecture 2.5.7 (Burr '80)**.** *Every oriented tree of order k is $(2k-2)$-universal.*

The existence of a $k(k-1)$ universality bound was shown by Burr, and improved to $(\frac{k^2}{2} - \frac{k}{2} + 1)$ by Addario-Berry, Havet, Linhares-Sales, Reed and Thomassé [Add+13]. Our main result is the first subquadratic bound for Burr's conjecture, in the following form (with $8\sqrt{2/15} \simeq 2.92$).

Theorem 3.1.1. *Every oriented tree of order k is $(8\sqrt{\frac{2}{15}}k\sqrt{k} + \frac{11}{3}k + \sqrt{\frac{5}{6}}\sqrt{k} + 1)$-universal.*

Moreover, we give improved functions for the case of arborescences, showing in Theorem 3.3.4 that they are $(\sqrt{4/3} \cdot k\sqrt{k} + k/2)$-universal (where $\sqrt{4/3} \simeq 1.15$).

Generalizing the famous Gallai-Roy-Hasse-Vitaver Theorem 2.5.2, saying $\overrightarrow{P_k}$ is k-universal, much research went into showing the Burr's conjecture, or at least linear bounds, for b-*blocks* paths. The conjecture has only been settled for 2-block paths, with a bound of k, by Addario-Berry et al. [AHT07], and linear bounds have been obtained for paths with 3 or 4 blocks [El 22, EG24]. We provide the first general linear bounds, with b fixed, for paths with b blocks.

Theorem 3.1.2. *For $b \geq 2$, every oriented path of order k with b blocks is $((b-1)(k-3)+3)$-universal.*

Along the way, this settles the conjecture for 3-block paths, giving us a bound of $2k-3$. For larger values, we improve this bound to $\frac{(b+3)}{2}k$ in Theorem 3.3.3.

3.1.1 Overview

Recall that both [Bur80] and [Add+13] showed quadratic universality bounds by induction on the order of trees, gluing (appending) leaves at each step to derive a new bound increasing only by k. We will extensively rely on the leaf gluing lemma from [Add+13], which we introduce in the following section. The cornerstone of our contributions is the establishment of such a gluing lemma appending oriented paths, and an improved version for directed paths. We leverage this in the following win-win strategy when deriving universality bounds for some T by induction. If T has many leaves (more than $O(\sqrt{k})$), we apply the gluing lemma of [Add+13] to the tree obtained by removing the leaves of T. Otherwise, we observe T is decomposable into few oriented paths, which we can glue one by one a limited number of times through our path gluing lemma.

In Section 3.2, we introduce various technical tools, and the gluing lemma for leaves. In Section 3.3, we prove our gluing lemma for directed paths. There, we derive our $\frac{(b+3)}{2}k$ and $(b-1)(k-3)+3$ bounds for b-block paths in Theorems 3.1.2 and 3.3.3, as well as the $\sqrt{4/3} \cdot k\sqrt{k} + O(k)$ bound for arborescences in Theorem 3.3.4. Then, in Section 3.4, we prove our gluing lemma for oriented paths and derive Theorem 3.1.1, showing that oriented trees of order k are $8\sqrt{2/15} \cdot k\sqrt{k} + O(k)$-universal. The proofs for arborescences and general oriented trees follow the same approach, with only the technical lemmas differing in nature.

3.2 Preliminaries

When clear from the context, we use "tree" or "path" to refer to an oriented tree or oriented path.

3.2.1 Rooted universality in DAGs

Given a digraph D, an *in-kernel* (respectively *out-kernel*) of D is an independent K set that in-dominates (respectively out-dominates) D. That is, every vertex in $V(D) - K$ admits an in-neighbour (respectively out-neighbour) in K. It is well-known that DAGs admit both an in-kernel and an out-kernel [vMR44].

The following lemma is an easy "rooted" extension of Theorem 2.4.8, from [Add+13]. For the sake of completeness, we provide its proof below.

Lemma 3.2.1. *Let T be an oriented tree of order k, and D be a directed acyclic graph such that $\chi(D) \geq k$, then D contains T. Moreover, rooting T in any vertex r, D admits a partition (X, K) of its vertex set with $\chi(D[K]) \leq k-1$ and such that for every vertex x of X, there exists a copy of T in D where r is identified to x and all other vertices of T lie in K.*

Proof. We prove the stronger second statement by induction on k. If $k = 1$, then we choose $X = V(D)$ and $K = \emptyset$. Now, consider $k > 1$ and a tree T on k vertices with root r. As $k > 1$, there exists a leaf l in T different from r. Denote $T \setminus \{l\}$ by T' and let l' be the (only) neighbour of l in T. Without loss of generality, we can assume that l' is an in-neighbour of l. Let K_0 be an in-kernel of D, which exists as D is acyclic. Since K_0 is independent, $\chi(D \setminus K_0) \geq k-1$, for otherwise colouring K_0 with a single colour would yield $\chi(D) \leq k-1$. Then, the induction hypothesis provides a partition (X', K') of $V(D \setminus K_0)$ for the tree T'. Now, let us show that $(X = X', K = K' \cup K_0)$ is the desired partition of D for T.

First, note that $\chi(D[K]) \leq \chi(D[K']) + \chi(D[K_0]) \leq (k-2) + 1 = k-1$. Moreover, for any vertex x in X, there exists a copy of T' in $D \setminus K_0$ where r is identified to x and the other vertices of T' lie in K'. In particular l' is in K' and as K_0 is an in-kernel of D, l' has an out-neighbour in K_0 to which we can identify l. □

3.2.2 Gluing leaves

We state the gluing lemma from [Add+13], improving on the one of Burr (Lemma 2.5.10). Let u be a leaf of T and consider u' its unique neighbour in T. Recall $Out(T)$ (resp. $In(T)$) refer to the set of all the out-leaves (resp. in-leaves) of T.

Lemma 3.2.2 (Addario-Berry et al. '13)**.** *For $k \geq 3$, let T be an oriented tree of order k other than S_k^+ or S_k^-. If $T - Out(T)$, respectively $T - In(T)$ is c-universal, then T is $(c + 2k - 4)$ universal.*

The next corollary follows by performing two gluing operations in a row, one for $Out(T)$ and one for $In(T)$.

Corollary 3.2.3. *For $k \geq 3$, let T be an oriented tree of order k other than S_k^+ or S_k^-. Let $L(T)$ denote the set of all leaves of T. If $T - L(T)$ is c-universal, then T is $(c + 4k - 9)$-universal.*

Proof. We may assume without loss of generality that T has at least one out-leaf, dealing with the case where T has at least in-leaf symmetrically. Our goal is to apply two gluing operations in a row, once for the out-leaves, then once for the in-leaves. Before being able to do so, we must take care of the case where $T - Out(T)$ is an in-star or an out-star. In this case, recall Burr's conjecture holds, and since T has an out-leaf, $|T - Out(T)| \leq k-1$, yielding that $T - Out(T)$ is $2(k-1) - 2 = 2k-4$-universal. In turn, Lemma 3.2.2 yields that T is $4k-8$-universal. Since $T - L(T)$ consists of a single vertex, which is 1-universal, T satisfies the lemma.

We may now assume $T' = T - Out(T)$ is neither an out-star nor an in-star. Then, we let T'' denote the tree $T' - In(T')$. Observe T'' is a subgraph of $T - L(T)$, so any digraph containing a copy of $T - L(T)$ contains a copy of T''. Therefore, T'' is c-universal, and since $|T'| \leq k-1$, Lemma 3.2.2 yields that T' is $(c+2k-6)$-universal. Applying the same lemma again on T, with $|T| = k$, we conclude that T is $(c+4k-10)$-universal, achieving to show the result. □

3.2.3 Decomposing trees into paths

The following decomposition of a tree into paths is rather classical, we prove it for the sake of completeness. Here, we state the lemma in terms of undirected trees. We will later apply it to oriented trees, and when doing so, we are implicitly decomposing the underlying tree into paths, and assume the paths given by the decomposition preserve their initial orientation. Let T be an undirected tree, rooted in some vertex r. A path $P = v_1, \dots, v_\ell$ of T is *descending* if v_i is the parent of v_{i+1} in T for $i = 1, \dots, \ell - 1$. Then, we consider v_1 as the *beginning* of P.

Lemma 3.2.4. *Let T be an undirected rooted tree with p leaves. Then, there exist p descending paths $P_1, \dots, P_p$ of T such that for every $i \in [1, p-1]$ the path P_{i+1} and $T_i = \bigcup_{j=1}^{i} P_j$ intersect in the vertex beginning P_{i+1}, and $T = T_p = \bigcup_{j=1}^{p} P_j$.*

Proof. Let us denote the leaves of T as $(l_1, ..., l_p)$. First, we define P_1 as the path from the root r of T to l_1, which contains no other leaf. Then, assume we have defined descending paths $P_1, \dots, P_i$ for $i < p$ satisfying the conditions of the lemma. Assume also that $T_i = \bigcup_{j=1}^{i} P_j$ contains exactly leaves $(l_j)_{j \leq i}$, and in particular $l_{i+1} \notin T_i$. We consider the descending path P'_{i+1} from r to l_{i+1}, noting $r \in T_i$, $l_{i+1} \notin T_i$, and T_i is a subtree of T. This allows us to define P_{i+1} as the subpath of P'_{i+1} beginning at the last vertex in T_i, and ending in l_{i+1}. Then, T_i and P_{i+1} only intersect in the vertex beginning P_{i+1}, and $T_{i+1} = T_i \cup P_{i+1}$ contains exactly $(l_j)_{j \leq i+1}$. When this process ends, we have defined p paths $P_1, \dots, P_p$, such that $T_p = \bigcup_{j=1}^{p} P_j = T$. □

3.3 Arborescences and b-block paths

In this section, we tackle the universality of arborescences and paths with b blocks. Consider a tree T' for which we have a universality bound c'. Our goal is to obtain a universality bound for the tree T obtained by gluing a directed path of length ℓ to some vertex in T'. It turns out that gluing directed paths as such is considerably cheaper, in terms of the increase on the bound, than gluing arbitrary orientations of paths. This gives us improved bounds for arborescences and b-blocks paths, and more generally for any oriented tree that decomposes into few directed paths.

3.3.1 Gluing a directed path

We first show a technical result, which provides us with all the structure required to prove the gluing lemma for directed paths.

Lemma 3.3.1. *For every integer $\ell \geq 0$, and every oriented graph D, there exists a partition of $V(D)$ into sets X, Y, Z such that:*

- (X) *Every vertex $x \in X$ admits both an in-neighbour y^- and an out-neighbour y^+ in Y,*
- (Y) *$D[Y]$ is a DAG, and every sink of $D[Y]$ is the beginning of a directed path of length ℓ whose remaining vertices are all in Z.*

(Z) $\chi(D[Z]) \leq \ell$.

Symmetrically, there also exists a partition (X, Y, Z) for D satisfying (X), (Z) and the following in place of (Y):

($\overleftarrow{\text{Y}}$) *$D[Y]$ is a DAG, and every source of $D[Y]$ is the endpoint of a directed path of length ℓ whose other vertices are all in Z.*

Proof. We show how to obtain the first partition, proceeding by induction on ℓ, which will also yield the result for the second partition. Indeed, to obtain the second partition, it suffices to reverse all the arcs of D and consider the first partition on the resulting digraph.

For $\ell = 0$, let $Z = \emptyset$, let Y be the vertex set of a maximal induced acyclic subdigraph of D, and let $X = V(D) \setminus Y$. Properties (Y) and (Z) hold trivially. For property (X), note that by maximality of Y, for any vertex $x \in X$ the graph $D[Y \cup \{x\}]$ contains a cycle passing through x. The neighbours y^-, y^+ of x on the cycle belong to Y, which yields property (X).

Assume by induction that the lemma holds for a fixed $\ell \geq 0$, and let us denote X', Y', Z' the corresponding subsets of $V(D)$. We will build X, Y, Z satisfying the lemma for $\ell + 1$, and refer the reader to Figure 3.1 for a sketch of the construction. Informally, in trying to satisfy (Y), we transfer sinks S' of Y' to Z', defining Z. Then, to satisfy (X), we transfer vertices from X' to the remaining vertices of Y', defining Y, after which X consists of the vertices left in X'. We then argue that the sinks S of Y admit an out-neighbour in S', allowing us to extend the directed path given by induction by an arc.

Let $S' \subseteq Y'$ be the set of sinks in $D[Y']$, shown in darker red, note that S' is a stable set in $D[Y']$, and thus also in D. Letting $Z = Z' \cup S'$, we then have $\chi(D[Z]) \leq \chi(D[Z']) + 1 \leq \ell + 1$, satisfying (Z).

We now turn to defining Y and X. Consider the subdigraph $D[Y' \setminus S']$, depicted in lighter green, which is acyclic since $D[Y']$ is. We define Y as a vertex set containing $Y' \setminus S'$, and which induces a maximal directed acyclic graph in $D[V \setminus Z]$. Then, we let $X = X' \setminus Y$. The set Y can be obtained by starting with $Y = Y'$ and iteratively adding vertices of X' (shown in darker green), while maintaining that $D[Y]$ is acyclic. After having considered all the vertices of X', we are guaranteed that for every $x \in X$, there is a cycle going through x in $D[Y \cup \{x\}]$. The neighbours y^-, y^+ of x on the cycle belong to Y, which yields property (X).

We now turn to property (Y), letting $S \subseteq Y$ be the set of sinks in $D[Y]$. Recall $Y = (Y' \setminus S') \cup (Y \cap X')$, and let us first show that $S \subseteq Y' \setminus S'$, meaning none of the sinks of $D[Y]$ belong to X'. Indeed, consider any $x' \in Y \cap X'$, and note the vertex $y'^+ \in Y'$ given by property (X') must belong to $Y' \setminus S'$. This is because of our assumption that D is oriented, so $y'^+ \neq y'^-$, and the $y'^+y'^-$-directed path then ensures that y'^+ is not a sink of $D[Y']$. Hence, x' has an out-neighbour y'^+ in $Y' \setminus S' \subseteq Y$, meaning it is not a sink of $D[Y]$.

We have just shown that sinks S of $D[Y]$ are sinks of $D[Y' \setminus S']$. Since these vertices are not sinks in $D[Y']$, they must have an out-neighbour in S'. By induction, all such out-neighbours are the beginning of a directed path of length ℓ whose remaining vertices are in Z'. Therefore, every vertex in S is the beginning of some directed path of length $\ell + 1$, whose remaining vertices are all in $Z = Z' \cup S'$, ensuring (Y). □

We are now ready to prove the gluing lemma for directed paths.

Lemma 3.3.2. *Let T' be an oriented tree of order $k' \geq 1$, and let T be the oriented tree obtained by appending a directed path of length ℓ from any of its endpoints to some vertex of T'. If T' is c'-universal, then T is $(c' + k' + 2\ell - 3)$-universal.*

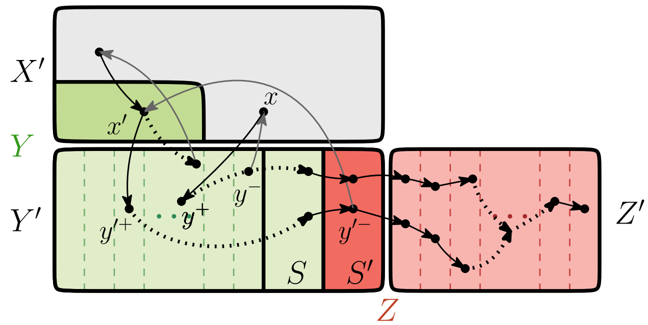


Figure 3.1: The partition of $V(D)$ into sets X', Y', Z'. The next step of the induction yields partition X, Y, Z, shown in grey, green, and red respectively. DAGs $D[Y']$ and $D[Y]$ are layered such that all arcs go from left to right. The sinks S' of $D[Y']$ begin a directed path of length ℓ continuing in Z', in dash-dotted. Then, the sinks S of $D[Y]$ begin a directed path of length $\ell+1$ continuing in Z.

Proof. Let T' be any oriented tree of order k', and consider a digraph D with chromatic number at least $c' + k' + 2\ell - 3$. We may assume D oriented, up to removing some arcs while preserving the chromatic number. We will first show how to append a directed path of length ℓ to T' by identifying the source of the path to some vertex of T'. Let us apply Lemma 3.3.1 to D with $\ell - 2$ in place of ℓ. We let X, Y, Z be the corresponding partition of $V(D)$, satisfying properties (X), (Y) and (Z). If $D[Y]$ has chromatic number at least $k' + \ell = |T|$, then by Lemma 3.2.1, it contains the tree T.

Otherwise, $D[Y]$ has chromatic number at most $k' + \ell - 1$. Since $D[Z]$ has chromatic number at most $\ell - 2$, $D[X]$ has chromatic number at least $c' + k' + 2\ell - 3 - (k' + \ell - 1) - (\ell - 2) = c'$. By hypothesis $D[X]$ must contain a copy of T', and we denote by x the vertex on this copy corresponding to the vertex of T' to which we wish to identify the source of the directed path. Then, property (X) ensures that x has an out-neighbour $y^+ \in Y$. If y^+ is not a sink, we may follow a directed path from y^+ towards a sink s of $D[Y]$, if it is, we let $y^+ = s$. Then, property (Y) yields a path P of length $\ell - 2$ starting in s and wholly contained in Z. Now, $(x, y^+, ..., s)$ along with the path P yield a directed path of length at least ℓ starting in x. Considering T' in X along with the path obtained from x yields the desired copy of T in D. The case is symmetrical when appending a directed path from its sink, using the second kind of partition and property $(\overleftarrow{\text{Y}})$ in Lemma 3.3.1. This achieves to show the lemma. □

3.3.2 Growing b-block paths

With the gluing lemma for directed paths in hand, the proof for b-block paths is now straightforward. We start with the bound for 2-block paths given by Theorem 2.5.4, and glue the remaining $b - 2$ paths one by one using Lemma 3.3.2.

Theorem 3.1.2. *For $b \geq 2$, every oriented path of order k with b blocks is $((b-1)(k-3)+3)$-universal.*

Proof. We proceed by induction on b. The case $b = 2$ holds by Theorem 2.5.4. We thus assume that P has $b \geq 3$ blocks, and let P' be a subpath of P obtained by deleting one of the end-blocks of P, that is, a block containing an end-vertex of P. We let ℓ be the length of this end-block, and note that P' has $k-\ell$ vertices.

By induction hypothesis, every graph with chromatic number at least $(b-2)(k-\ell-3)+3$ contains a copy of P'. Now, P is obtained from P' by appending a directed path of length ℓ. We apply Lemma 3.3.2, which yields that P is $((b-2)(k-\ell-3)+3+(k-\ell)+2\ell-3)$-universal. Then, we have $(b-2)(k-\ell-3)+3+(k-\ell)+2\ell-3 \leq (b-1)(k-\ell-3)+2\ell+3$. Since $2\ell+3 \leq (b-1)\ell+3$ as $b \geq 3$, we finally obtain $(b-1)(k-\ell-3)+2\ell+3 \leq (b-1)(k-3)+3$. This achieves to show that paths of order k with b blocks are $((b-1)(k-3)+3)$ universal. □

3.3.3 An improved bound for b-block paths

Theorem 3.3.3. *Every oriented path of order k with b blocks is $\frac{(b+3)}{2}k$-universal.*

Proof. The result already hold for $b \leq 2$ by Theorem 2.5.4. Assume then we are given an oriented path P with $b \geq 2$ blocks $P_1, ..., P_b$ appearing in order. We consider two ways of gluing these blocks, starting either from P_1 or from P_b. The subpath of P consisting of the first two blocks is $|P_1|+|P_2|$-universal. Then, we apply Lemma 3.3.2 successively to glue paths $(P_i)_i$ for i increasing from 3 to b. This yields that P is universal for the following value:

$$\begin{aligned}
&|P_1|+|P_2|+\sum_{i=3}^{b}\left(-3+2|P_i|+\sum_{j=1}^{i-1}|P_j|\right)\\
=&-3b+|P_1|+|P_2|+2\sum_{i=3}^{b}|P_i|+\sum_{i=3}^{b}\sum_{j=1}^{i-1}|P_j|\\
\leq&-3b+|P_1|+2\sum_{i=2}^{b}|P_i|+\sum_{i=2}^{b-1}\sum_{j=1}^{i}|P_j|\\
\leq&-3b+2k+\sum_{i=1}^{b-1}\sum_{j=1}^{i}|P_j|
\end{aligned}$$

The same reasoning starting with the last two blocks and gluing for i decreasing from $b-2$ to 1 yields that P is also universal for the following value:

$$-3b+2\sum_{i=2}^{b}|P_{b-i+1}|+\sum_{i=1}^{b-1}\sum_{j=1}^{i}|P_{b-j+1}|$$

Now, summing the last term of each equation yields:

$$\begin{aligned}
&\sum_{i=1}^{b-1}\sum_{j=1}^{i}|P_j| + \sum_{i=1}^{b-1}\sum_{j=1}^{i}|P_{b-j+1}| \\
=&\sum_{i=1}^{b-1}\sum_{j=1}^{i}|P_j| + \sum_{i=1}^{b-1}\sum_{j=b-i+1}^{b}|P_j| \\
=&\sum_{i=1}^{b-1}\sum_{j=1}^{i}|P_j| + \sum_{i=2}^{b}\sum_{j=i}^{b}|P_j| \\
=&\sum_{i=1}^{b-1}\sum_{j=1}^{i}|P_j| + \sum_{i=1}^{b-1}\sum_{j=i+1}^{b}|P_j| \\
=&(b-1)k
\end{aligned}$$

Therefore, one of the sums is at most $\frac{(b-1)}{2k}$, and adding the remaining $-3b+2k$ term yields that P is $(\frac{(b+3)k}{2} - 3b)$-universal. □

Note that taking paths on $b+1$ blocks, made up of an arbitrarily large middle directed path, with alternating paths of length $b/2$ glued on each end, yields that the bound is asymptotically tight for this strategy.

3.3.4 Growing arborescences

We now move on to the case of arborescences. The proof is by induction, and follows a win-win argument that will also be used in the general case. At a given step, if there are more than $O(\sqrt{k})$ leaves, we use the gluing lemma for leaves, otherwise we leverage our gluing lemma for directed paths.

Theorem 3.3.4. *Every k-vertex arborescence is $(\sqrt{\frac{4}{3}}k\sqrt{k} + \frac{k}{2})$-universal.*

Proof. Let us show the result for out-arborescences, the case of in-arborescences is symmetrical. We set $g(k) = \sqrt{4/3} \cdot k\sqrt{k} + k/2$. First, notice that the statement holds trivially for $k = 1$ and $k = 2$, as we have $g(1) \geq 1$ and $g(2) \geq 2$. Then, if T is an out-star S_k^+, recall it is $2k-2$-universal, and since $2k - 2 \leq g(k)$, it is $g(k)$-universal.

Let us proceed by induction on k, the order of T. By the above, we can assume that $k \geq 3$ and take any out-arborescences of order k distinct from S_k^+. We distinguish two cases to show that T is $f(k)$-universal, according to the number p of (out-)leaves in T. We first settle the case $p \geq \sqrt{4k/3}$, then deal with the case $p < \sqrt{4k/3}$.

Assume first that T contains at least $\sqrt{4k/3}$ (out-)leaves. Let T' be the tree obtained from T by removing its leaves, that is $T' = T - Out(T)$, and note $|T'| \leq k - \sqrt{4k/3}$. By induction, we know that T' is $g(k - \sqrt{4k/3})$-universal. Then, recall $T \neq S_k^+$, so Lemma 3.2.2 yields that T is $g(k - \sqrt{4k/3}) + 2k - 4$-universal. To show that T is $g(k)$-universal, it suffices to verify

$g(k) \geq g(k - \sqrt{4k/3}) + 2k - 4$, that is:

$$\sqrt{\frac{4}{3}}\left(k - \sqrt{\frac{4k}{3}}\right)^{\frac{3}{2}} + \frac{k - \sqrt{4k/3}}{2} \leq \sqrt{\frac{4}{3}}k\sqrt{k} + \frac{k}{2} \qquad \forall k \geq 2 \tag{3.1}$$

which we numerically verify for all $k \geq 3$.

From now on, we may therefore assume T contains $p < \sqrt{4k/3}$ leaves. Using Lemma 3.2.4 on the tree underlying T rooted in an arbitrary vertex, we obtain p descending paths $P_1, \dots, P_p$. We consider these paths with their initial orientation, and as T is an out-arborescence, each P_i is a directed path. We let $T_i = \bigcup_{j=1}^{i} P_j$, such that $T = T_p$, and we know that T_i and P_{i+1} only intersect at the beginning of P_{i+1}.

Now, we show by induction on i that T_i is $(i \cdot |T_i|)$-universal. For $i = 1$, as T_1 is a directed path, it is $|T_1|$-universal by the Gallai-Roy-Hasse-Vitaver Theorem. Assume the induction hypothesis holds for some $i < p$. Consider T_{i+1}, which is obtained from T_i by appending the path P_{i+1} rooted at their intersecting vertex. Then, Lemma 3.3.2 yields that T_{i+1} is $(i \cdot |T_i| + |T_i| + 2l_{i+1} - 3)$-universal, where l_{i+1} is the length of P_{i+1}. As $|T_{i+1}| = |T_i| + l_{i+1}$, we obtain $i \cdot |T_i| + |T_i| + 2l_{i+1} - 3 \leq (i-1) \cdot |T_i| + 2(|T_i| + l_{i+1}) \leq (i-1) \cdot |T_{i+1}| + 2(|T_{i+1}|) = (i+1) \cdot |T_{i+1}|$. Thus, we conclude that T_{i+1} is $((i+1) \cdot |T_{i+1}|)$-universal, proving step $i+1$ of the induction. Finally, this yields that $T = T_p$ is pk-universal, and $pk < \sqrt{4k/3} \cdot k$. Since $g(k) > \sqrt{4/3} \cdot k\sqrt{k}$, this achieves to show that T is $g(k)$-universal, concluding the proof. □

3.4 Oriented trees

In this section, we obtain universality bounds for general oriented trees. As in the case of arborescences, we consider an oriented tree T', for which we have a bound c'. Then, we derive a universality bound for the tree T obtained by appending an oriented path Q to T'. We adapt the techniques of the previous section from directed paths to general oriented paths.

3.4.1 Gluing an oriented path

Given an oriented path Q, we obtain a *rooted oriented path* from Q by choosing one of the extremities of Q as the *root* of Q. Now, given a rooted oriented path Q, with root r, a digraph D and a vertex x of D, we say that D contains a copy of Q *starting at* x if D contains a copy of Q where r is identified to x.

The next two lemmas are the respective counterparts of Lemma 3.3.1 and Lemma 3.3.2. In both of those, the obtained bounds now depend quadratically on the length of the path being appended, where this dependency was linear in the last section.

Lemma 3.4.1. *Let $\ell \geq 0$ be an integer, and Q a rooted oriented path of length ℓ. Then, for every oriented graph D, there exists a partition of $V(D)$ into sets X, Y, Z such that:*

(X) *Every vertex $x \in X$ admits both an in-neighbour y^- and an out-neighbour y^+ in Y,*

(Y) *Y induces a directed acyclic graph, and for every vertex $y \in Y$, there exists a copy of Q starting at y and contained in $D[Y \cup Z]$,*

(Z) $\chi(D[Z]) \leq \frac{\ell(\ell+1)}{2}$.

Proof. We prove the result by induction on ℓ. For $\ell = 0$, let $Z = \emptyset$, let Y be the vertex set of a maximal acyclic subdigraph of D, and $X = V(D) \setminus Y$. Properties (Y) and (Z) hold trivially, while property (X) follows from the maximality of $D[Y]$ as an acyclic subdigraph of D.

Assume by induction that the lemma holds for any rooted oriented path of length $\ell \geq 0$. To show it holds for paths of length $\ell + 1$, let us consider any oriented path Q of length $\ell + 1$ rooted in r. Let r' be the unique neighbour of r in Q and let Q' be the oriented path $Q - r$ rooted in r'. Consider the partition $X', Y', Z' \subseteq V(D)$ given by induction to satisfy the properties of the lemma for Q'. We will build partition X, Y, Z satisfying the properties for Q, and refer the reader to Figure 3.2 for a sketch. Informally, in trying to satisfy (Y), we transfer the vertices of a suitably chosen set $K' \subseteq Y'$, from Y' to Z', defining Z. Then, to satisfy (X), we transfer vertices from X' to the remaining vertices of Y', defining Y, after which X consists of the vertices left in X'. We show how to build copies of Q starting in Y, either thanks to K', or by appending a neighbour to some copy of Q' obtained by induction.

Let us start by defining Z, exhibiting a set $K' \subseteq Y'$ such that $Z = Z' \cup K'$. If $\chi(D[Y']) \leq \ell+1$, we simply define $K' = Y'$. Otherwise, $D[Y']$ is an acyclic digraph such that $\chi(D[Y']) \geq \ell + 2$. We may now apply Lemma 3.2.1 to $D[Y']$, with Q as the oriented tree T rooted in r (note that $|Q| = \ell + 2$). This gives us a subset $K' \subseteq Y'$ (shown in darker red), with $\chi(D[K']) \leq \ell + 1$, and such that every $y \in Y' \setminus K$ is the beginning of some Q whose remaining vertices belong to K'. We let $Z = Z' \cup K'$. We have shown that $\chi(D[K']) \leq \ell + 1$ in both cases above, and by induction we have $\chi(D[Z']) \leq \frac{\ell(\ell+1)}{2}$. Colouring $D[K']$ and $D[Z']$ with a different set of colours gives $\chi(D[Z]) \leq \frac{\ell(\ell+1)}{2} + (\ell + 1) = \frac{(\ell+1)(\ell+2)}{2}$, satisfying (Z).

We now turn to defining Y and X. Consider the (possibly empty) subdigraph $D[Y' \setminus K']$, depicted in lighter green, which is acyclic since $D[Y']$ is. We define Y as a vertex set containing $Y' \setminus K'$, and which induces a maximal directed acyclic graph in $D[V \setminus Z]$. Then, we let $X = X' \setminus Y$. The set Y can be obtained by starting with $Y = Y'$ and iteratively adding vertices of X' (shown in darker green), while maintaining that $D[Y]$ is acyclic. After having considered all the vertices of X', we are guaranteed that for every $x \in X$, there is a cycle going through x in $D[Y \cup \{x\}]$. The neighbours y^-, y^+ of x on the cycle belong to Y, which yields property (X).

It remains to show that property (Y) holds. Recall $Y = (Y' \setminus K') \cup (Y \cap X')$, to build Q starting from any vertex of Y, we use two strategies by considering either $y \in (Y' \setminus K')$ or $x' \in (Y \cap X')$. In the first case, $Y' \setminus K' \neq \emptyset$, and we already argued that $\chi(D[Y']) \geq \ell + 2$. Then, recall y must be the beginning of a subpath Q whose remaining vertices lie in K'. Since $y \in Y$ and $K' \subseteq Z$, Q is contained in $D[Y \cup Z]$, ensuring (Y) here. In the second case, we have $x' \in X'$ in particular, so x' admits both an in-neighbour y'^- and an out-neighbour y'^+ in Y' by induction. Now, Q may be obtained from Q' by appending either an out-neighbour, or respectively an in-neighbour, to its root r'. By induction, we may then consider a copy of Q' starting at either y'^-, or y'^+ respectively. See two examples for y'^- in Figure 3.2. This copy is contained in $D[Y' \cup Z']$, and in particular it cannot contain x' since $(Y \cap X') \cap ((Y' \setminus K') \cup Z') = \emptyset$. Therefore, (x', y'^-), or (x', y'^+) respectively, along with the copy of Q' yields a copy of Q starting in x', achieving to show (Y). □

We now move on to show the gluing lemma for oriented paths.

Lemma 3.4.2. *Let T' be an oriented tree of order $k' \geq 1$, and let T be the oriented tree obtained by appending a rooted oriented path of length ℓ to any vertex of T'. If T' is c'-universal, then T is $(c' + k' + \frac{\ell(\ell+1)}{2} - 1)$-universal.*

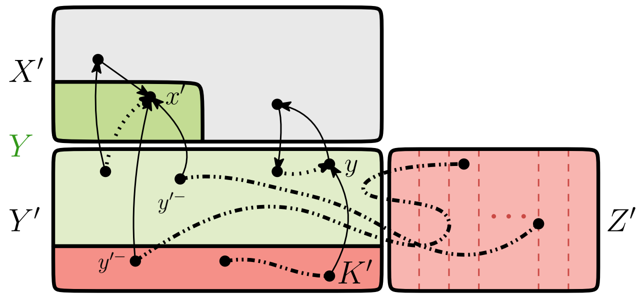


Figure 3.2: The partition of $V(D)$ into sets X', Y', Z'. The next step of the induction yields partition X, Y, Z, shown in grey, green, and red respectively. Vertices of Y' begin a copy of Q', in dash-dotted. Then, vertices of Y begin a copy of Q, either in K' (see y), or by appending an arc to some Q' (see x').

Proof. Let us consider a digraph D with chromatic number at least $c' + k' + \ell(\ell+1)/2 - 1$, which we may assume to be oriented without loss of generality (as this preserves the chromatic number). Let Q be an oriented path of length ℓ rooted at r, which we want to append to T. Let r' be the neighbour of r in Q, and let Q' be the oriented path of length $\ell - 1$ rooted in r' obtained by deleting r from Q. We start by applying Lemma 3.4.1 to D with the rooted path Q'. We let X, Y, Z be the corresponding partition of $V(D)$, satisfying properties (X), (Y) and (Z). If $D[Y]$ has chromatic number at least $k' + \ell = |T|$, since it is a directed acyclic graph, it contains the tree T by Lemma 3.2.1.

Otherwise, $\chi(D[Y]) \leq k' + \ell - 1$, and we also know $\chi(D[Z]) \leq (\ell-1)\ell/2$ by property (Z). Therefore, $D[X]$ has chromatic number at least $(c' + k' + \ell(\ell+1)/2 - 1) - (k' + \ell - 1) - (\ell-1)\ell/2 = c'$. By hypothesis, $D[X]$ must contain a copy of T', which we identify with T' implicitly, and we denote x the vertex on this copy where Q has to start to form T. Then, property (X) ensures us that x has both an in-neighbour y^- and an out-neighbour y^+ in Y. Both y^- and y^+ are the beginning of a rooted copy of Q' contained in $Y \cup Z$. Therefore, according to the orientation of the arc between r and r' in Q, we obtain a rooted copy of Q starting in x with either (y^-, x) or (y, x^+), with all remaining vertices in $Y \cup Z$. Except from x, this copy of Q is disjoint from the copy of T' obtained in x, yielding a copy of T in D. □

3.4.2 Growing oriented trees

We are now ready to prove our main result, bounding the universality of oriented trees. As for the case of arborescences, we proceed by induction, and use the same win-win argument. If there are more than $O(\sqrt{k})$ leaves, we glue leaves through Corollary 3.2.3, otherwise we glue oriented paths through Lemma 3.4.2.

Theorem 3.1.1. *Every oriented tree of order k is $(8\sqrt{\frac{2}{15}}k\sqrt{k} + \frac{11}{3}k + \sqrt{\frac{5}{6}}\sqrt{k} + 1)$-universal.*

Proof. We set $f(k) = 8\sqrt{2/15} \cdot k\sqrt{k} + 11k/3 + \sqrt{5/6} \cdot \sqrt{k} + 1$. First, notice that the statement holds trivially for $k = 1$ and $k = 2$, as we have $f(1) \geq 1$ and $f(2) \geq 2$. On the other hand, if T

is an out-star S_k^+ or an in-star S_k^-, recall it is $(2k-2)$-universal, and since $2k-2 \leq f(k)$, it is $f(k)$-universal.

Let us now proceed by induction on k, the order of T. By the above, we can assume $k \geq 3$, and take any oriented tree T of order k distinct from S_k^+ and S_k^-. We show T is $f(k)$-universal by using two strategies according to the number p of leaves in T. We first settle the case $p \geq \sqrt{5k/6}$, then deal with the case $p < \sqrt{5k/6}$.

Assume first that T contains at least $\sqrt{5k/6}$ leaves. Consider the subtree T' of T obtained by removing all of its leaves, and note $|T'| \leq k - \sqrt{5k/6}$. By induction, we know that T' is $f(k-\sqrt{5k/6})$-universal, then Corollary 3.2.3 yields that T is $(f(k-5k/6)+4k-9)$-universal. To show that T is $f(k)$-universal, it suffices to verify $f(k) \geq f(k-\sqrt{5k/6})+4k-9$, that is:

$$\begin{aligned}&\frac{8\sqrt{30}}{15}k^{3/2}+\frac{11}{3}k+\sqrt{\frac{5}{6}}\sqrt{k}+1\\ \geq&\frac{8\sqrt{30}}{15}(k-\sqrt{\frac{5}{6}k})^{3/2}+\frac{11}{3}(k-\sqrt{\frac{5}{6}k})+\sqrt{\frac{5}{6}}\sqrt{k-\sqrt{\frac{5}{6}k}}+4k-8\end{aligned}$$

which we numerically verify for all $k \geq 3$.

From now on, we may therefore assume that T contains $p < \sqrt{5k/6}$ leaves. Our goal is to split T into oriented paths, which we use to build T by using Lemma 3.4.2. Let us first apply Lemma 3.2.4 on the tree underlying T, rooted in an arbitrary vertex r. This yields descending paths $Q_1, \ldots, Q_p$, which we root in their endpoint closest to r. We consider these rooted paths with their initial orientation, such that $T = T_p = \bigcup_{j=1}^p Q_j$. Note then that T can be obtained recursively by starting from $T_1 = Q_1$, and building T_{i+1} by identifying the root of Q_{i+1} to some vertex of T_i.

At this point, the number of paths is bounded by p, but in order to apply Lemma 3.4.2 successfully, we also need to control their length. To do so, we cut paths of length exceeding $\ell = \lceil\sqrt{6k/5}\rceil \leq \sqrt{6k/5}+1$ into smaller ones as follows. For $j \in [1,p]$, we split Q_j into a minimum number of consecutive subpaths $(Q_{j,h})_h$, such that all have length exactly ℓ except for possibly the last one, which may have length strictly less than ℓ. We denote $(P_i)_{i\in[1,m]}$ the m resulting paths, and order them according to $(Q_j)_j$ they stem from, then with respect to their distance to the root of Q_j. Observe that T can still be constructed recursively using $(P_i)_i$, starting with $T_1 = P_1$, and constructing T_{i+1} by appending P_{i+1} to T_i. Then, we have $T = T_m = \bigcup_{i=1}^m P_i$.

To bound the number m of newly created paths $(P_i)_i$, we count those of length less than ℓ separately from those with length exactly ℓ. By definition, at most one P_i of length strictly less than ℓ is created per Q_j, so there are at most $p < \sqrt{5k/6}$ of them. Let us now bound the number of paths of length exactly $\ell = \lceil\sqrt{6k/5}\rceil$. Note T consists of exactly $k-1$ arcs, and since the paths $(P_i)_i$ are arc-disjoint, there must be at most $\lfloor(k-1)/\lceil\sqrt{6k/5}\rceil\rfloor \leq \sqrt{5k/6}$ of them. Finally, $(P_i)_i$ consists of $m \leq 2\sqrt{5k/6}$ paths, each of length at most $\ell \leq \sqrt{6k/5}+1$.

Recall that $T = T_m$. We are now ready to show that T is $f(k)$-universal, which we do by recursively computing bounds $(c_i)_i$ such that T_i is c_i-universal. Assume T_i, of order k_i, is c_i-universal. Recall T_{i+1} is obtained from T_i by appending P_{i+1}, which has length at most ℓ. We apply Lemma 3.4.2, which yields that T_{i+1} is c_{i+1}-universal for $c_{i+1} = c_i + k_i + \ell(\ell+1)/2 - 1$. Now, since T_i is obtained from P_1, of order at most $\ell+1$, by appending $(i-1)$ paths of length at most

ℓ, we have $k_i \leq i\ell + 1$. This yields:

$$c_{i+1} \leq c_i + i\ell + 1 + \frac{\ell(\ell+1)}{2} - 1 = c_i + i\ell + \frac{\ell(\ell+1)}{2}$$
$$c_{i+1} \leq c_1 + \ell(\sum_{j=1}^{i} j) + \frac{i\ell(\ell+1)}{2} \quad \text{by an immediate induction}$$
$$c_{i+1} \leq c_1 + \frac{1}{2}\ell i(\ell + i + 2)$$

Now, since $|T_1| \leq \ell + 1$ we have $c_1 \leq \ell(\ell+1)/2 + 1$ by Theorem 2.5.8. We set $i = m - 1$, and with the inequalities $\ell \leq \sqrt{6k/5} + 1$ and $m \leq 2\sqrt{5k/6}$ we obtain:

$$c_m \leq \frac{\ell(\ell+1)}{2} + 1 + \frac{1}{2}\ell(m-1)(\ell + m + 1)$$
$$c_m \leq \frac{\ell m}{2}(\ell + m) + 1$$
$$c_m \leq \frac{1}{2}(\sqrt{\frac{6}{5}k} + 1)(2\sqrt{\frac{5}{6}k})\Big(\sqrt{\frac{6}{5}k} + 1 + 2\sqrt{\frac{5}{6}k}\Big) + 1$$
$$c_m \leq 8\sqrt{\frac{2}{15}}k^{3/2} + \frac{11}{3}k + \sqrt{\frac{5}{6}}\sqrt{k} + 1$$

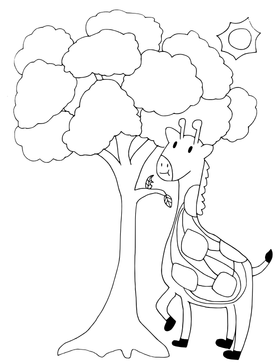
This is exactly saying $c_m \leq f(k)$, which achieves to prove that T is $f(k)$-universal and concludes the proof. □

3.5 Further directions

Our ability to control the growth of our universality bound for trees as $O(k\sqrt{k})$ stems from our construction of trees, and corresponding bounds, by gluing both paths and leaves. In the border cases where a tree can be built by adding many leaves at each step, or by gluing few paths, better bounds can be obtained. See Proposition 8 in [Add+13] for leaves, which can be seen as orthogonal to our bound for b-block paths. Nevertheless, each gluing operation to some tree T increases the bound by at least $|T|$. Thus, if a tree falls between these two categories, such as a star of degree $\sqrt{k}$ subdivided $\sqrt{k}$ times, our current techniques do not allow us to improve on the $O(k\sqrt{k})$ bound. Adapting one of the gluing lemmas to get rid of this $O(|T|)$ cost may allow us to skew our win-win strategy towards that type of gluing, and yield a better bound. Another way to improve our strategy is to manage to glue multiple paths in parallel, or even more general trees. Note that our cost for gluing a path depends quadratically on its order, so we are still far from a linear bound for paths. Nevertheless, even an improvement in that direction would not directly yield bounds better than $O(k\sqrt{k})$ for the general case, as witnessed by the case of arborescences.

Chapter 4

A directed Gyárfás-Sumner for orientations of $P4$



This chapter concerns our results on a directed analogue of the Gyárfás-Sumner conjecture [ACN21], introduced in Section 2.6. It is based on joint work with Linda Cook, Tomas Masarik, Marcin Pilipczuk, and Ueverton S. Souza, appearing in [Coo+23].

4.1 Introduction

A class of digraphs $\mathcal{D}$ is $\vec{\chi}$-*bounded* if there exists a function f such that every $D \in \mathcal{D}$ satisfies $\vec{\chi}(D) \leq f(\omega(D))$. Then, the extension of Gyárfás-Sumner to digraphs proposed by Aboulker, Charbit and Naserasr is the following [ACN21].

Conjecture 4.1.1 (The ACN $\vec{\chi}$-boundedness conjecture [ACN21])**.** *Every oriented forest is $\vec{\chi}$-bounding.*

Recall the conjecture is known for oriented stars, thus orientations of P_3, and we are here concerned with orientations of P_4. There are four different orientations of P_4, up to reversing the order of the vertices.

- The directed orientation $\rightarrow\rightarrow\rightarrow$ is denoted $\overrightarrow{P_4}$.
- The alternating orientation $\leftarrow\rightarrow\leftarrow$ is denoted $\overrightarrow{A_4}$.
- We denote $\rightarrow\leftarrow\leftarrow$ and $\leftarrow\leftarrow\rightarrow$ by $\overrightarrow{Q_4}$ and $\overrightarrow{Q_4}'$, respectively.

Note that $\overrightarrow{Q_4}$ and $\overrightarrow{Q_4}'$ can be obtained from each other by reversing the direction of every edge in the path, meaning one is $\vec{\chi}$-bounded if and only if the other is. For these two orientations, the Conjecture 4.1.1 follows from their χ-boundedness [CS14], meaning the two open cases are $\overrightarrow{P_4}$ and $\overrightarrow{A_4}$.

In this chapter, we prove $\vec{\chi}$-boundedness for all orientations, through a common strategy, settling the conjecture for $\overrightarrow{P_4}$ and $\overrightarrow{A_4}$ as well.

Theorem 4.1.2. *Let H be any orientation of P_4, then the class of H-free oriented graphs is $\vec{\chi}$-bounded. In particular, for any H-free oriented graph D,*

$$\vec{\chi}(D) \leq (\omega(D) + 7)^{(\omega(D)+8.5)}.$$

We include in a new proof that $\overrightarrow{Q_4}$ and $\overrightarrow{Q_4}'$ are both $\vec{\chi}$-bounding and improve the $\vec{\chi}$-binding function for the classes of $\overrightarrow{Q_4}$-free oriented graphs and $\overrightarrow{Q_4}'$-free oriented graphs.

Our result also answers a question raised by Steiner [Ste23] in the affirmative. That is, for $H \in \{\overrightarrow{P_4}, \overrightarrow{A_4}\}$ and any $k \geq 4$ the class of H-free oriented graphs not containing a transitive tournament of order k has bounded dichromatic number. Note that after our paper, the result was shown for $\overrightarrow{P_6}$ in the case of triangle-free digraphs [Abo+24], but all other cases remain open.

4.1.1 Structure of the chapter and proof overview.

Let H be any orientation of P_4. We prove Theorem 4.1.2 by induction on the clique number. We fix an integer $\omega(D) \geq 2$. We define a function f and assume that H-free oriented graphs with clique number ω' where $1 \leq \omega' < \omega$ have dichromatic number at most $f(\omega')$. We then consider an oriented graph with clique number ω and show that D can be dicoloured using at most $f(\omega)$ colours.

Our strategy to bound $\vec{\chi}(D)$ crucially relies on a tool we call *dipolar sets* which were introduced by the name "nice sets" in [ACN21].

Definition 4.1.3 (*dipolar set*)**.** A *dipolar set* of an oriented graph D is a nonempty subset $S \subseteq V(D)$ that can be partitioned into S^+, S^- such that no vertex in S^+ has an out-neighbour in $V(D \setminus S)$ and no vertex in S^- has an in-neighbour in $V(D \setminus S)$.

Dipolar sets have the following useful property [ACN21]: In order to bound the dichromatic number of a class of oriented graphs closed under taking induced subgraphs, it suffices to exhibit a dipolar set of bounded dichromatic number for each of the members in the class. We give a few preliminary observations as well as an introduction to dipolar sets in Section 4.2.

In Section 4.3, we show how to construct a dipolar set for any H-free oriented graph D of clique number ω. Our goal is to obtain a bound for the dichromatic number of this set. The backbone of our construction is an object we call a closed tournament.

Definition 4.1.4 (path-minimizing closed tournament)**.** We say K and P *form a closed tournament* $C = K \cup V(P)$ if K is a tournament of order $\omega(D)$ and P is a directed path from a sink component to a source component in K.

Given a digraph D, we say $C = K \cup P$ forms a *path-minimizing closed tournament* if it is a closed tournament such that K is a maximum clique of D, and $|P|$ is minimized amongst all possible choices of K, P that form a closed tournament.

It follows from the definition of closed tournament that the graph induced by a closed tournament is strongly connected and that every strongly connected oriented graph has a path-minimizing closed tournament. We will define a set S consisting of the closed neighbourhood of a path-minimizing closed tournament C and a subset of the second neighbours of C. We will show that if D is H-free, then S is a dipolar set. This proof will rely heavily on the fact that C is strongly connected.

The strong connectivity of C is a powerful property in showing that S is a dipolar set. However, ensuring C is strongly connected by adding P to K makes it harder to bound the dichromatic number of $N(C)$. We explain in Section 4.2 that we can easily bind the dichromatic number of the first neighbourhood of any bounded cardinality set. Unfortunately, we have no control over the cardinality of P in a path-minimum closed tournament. In fact, P, and thus C, might be arbitrarily large with respect to ω. This makes the task of bounding the dichromatic number of $N(C)$ significantly harder. Fortunately, since D is H-free and we may choose C to be a *path-minimizing* closed tournament, there are a lot of restrictions on what arcs may exist between vertices of $N(C)$. Ultimately, our goal is to exploit these restrictions to bind the dichromatic number of $N(C)$.

Interestingly, we can define S and prove that it is a dipolar set in the same way for each possible choice of an oriented P_4. We describe our construction of a dipolar set S in Section 4.3. However, we used different (but similar) proofs to show that S has bounded dichromatic number for $H = \overrightarrow{P_4}$, $\overrightarrow{A_4}$, and $\overrightarrow{Q_4}$. The proof that S has bounded dichromatic number when H is $\overrightarrow{Q_4}$ implies the result when H is $\overrightarrow{Q_4}'$.

In Section 4.4, we bound the dichromatic number of C, the vertices of S in the second neighbourhood of C, and $N(K)$ for H-free graphs where H is an arbitrary choice of an orientated P_4. In Section 4.5, we bound the dichromatic number of the vertices in S not handled in Section 4.4. These remaining vertices are the set $N(P) \setminus N[K]$. Here we use separate (but similar) proofs for $H = \overrightarrow{P_4}, \overrightarrow{A_4}, \overrightarrow{Q_4}$. In Section 4.6, we put the pieces together to obtain our main result that any orientation of P_4 is $\overrightarrow{\chi}$-bounding. We discuss some related open questions in Section 4.7.

4.2 Preliminaries

In this section, we lay the groundwork for our proof by making a few observations useful in later sections and introducing dipolar sets. In the rest of the chapter, we will only consider strongly connected oriented graphs since the dichromatic number of an oriented graph is equal to the maximum dichromatic number of one of its strongly connected components.

4.2.1 Definitions

Throughout this chapter, we say that a (di)graph G *contains* a (di)graph H if G contains H as an *induced* sub(di)graph. All digraphs considered here are oriented. For a subdigraph $H \subseteq D$ we let $N(H)$ denote the set $N(V(H))$. If G does not contain a (di)graph H we say that G is *H-free*. If G does not contain any of the (di)graphs $H_1, H_2, \dots, H_k$ we say G is *$(H_1, H_2, \dots, H_k)$-free*. Recall the *clique number* $\omega(D)$ of a digraph D is the clique number of its underlying graph. If a class $\mathcal{D}$ is *$\overrightarrow{\chi}$-bounded* by a function f, meaning every $D \in \mathcal{D}$ satisfies $\overrightarrow{\chi}(D) \leq f(\omega(D))$, we say f a *$\overrightarrow{\chi}$-binding function* for $\mathcal{D}$.

4.2.2 Induction hypothesis

We will work with the following assumptions:

Scenario 4.2.1 (Inductive Hypothesis)**.** *Let H be an oriented P_4 and let $\omega > 1$ be an integer. We let γ be the maximum of $\overrightarrow{\chi}(D')$ over every H-free oriented graph D' satisfying $\omega(D') < \omega$, and assume γ is finite. We let D be an H-free oriented graph with clique number ω and assume D is strongly connected.*

We will aim to bound the $\overrightarrow{\chi}(D)$ in terms of γ and ω. We begin with some easy observations about the dichromatic number of the neighbourhood of any sets of vertices in D. For any vertex $v \in V(D)$, by definition $\omega(N(v)) \leq \omega - 1$ as otherwise D would contain a tournament of size greater than ω. Hence, for any $v \in V(D)$, $\overrightarrow{\chi}(N(v)) \leq \gamma$. This can be directly extended to bounding the dichromatic number of the neighbourhood of a set of a given size as follows:

Observation 4.2.2. *Let D be an oriented graph and let γ be the maximum value of $\overrightarrow{\chi}(N(v))$ for any $v \in V(D)$. Then every $X \subseteq V(D)$ satisfies:*

$$\overrightarrow{\chi}(N(X)) \leq \overrightarrow{\chi}\left(\bigcup_{x \in X} N(x)\right) \leq |X| \cdot \gamma.$$

4.2.3 Using dipolar sets

We now state the main property of dipolar sets, shown in [ACN21]. The following reduces the task of bounding the dichromatic number of D, to that of bounding the dichromatic number of a dipolar set for every induced oriented subgraph of D.

Lemma 4.2.3 (Lemma 17 in [ACN21])**.** *Let $\mathcal{D}$ be a family of oriented graphs closed under taking induced subgraphs. Suppose there exists a constant c such that every $D \in \mathcal{D}$ has a dipolar set S with $\overrightarrow{\chi}(S) \leq c$. Then every $D \in \mathcal{D}$ satisfies $\overrightarrow{\chi}(D) \leq 2c$.*

Proof. Consider such a family $\mathcal{D}$, and let D be a minimal counter-example to the lemma. Let S be a dipolar set in D, and $D' = D - S$. Take the partition of S into S^+, S^-, such that S^+ has no out-neighbour in D', and S^- has no in-neighbours in D'. By minimality of D', it can be coloured using at most $2c$ colours, and we now show how to extend this colouring to D. Since $\overrightarrow{\chi}(S) \leq c$, we also have $\overrightarrow{\chi}(S^+), \overrightarrow{\chi}(S^-) \leq c$. Then, properly colour S^+ and S^- with disjoint palettes of c colours each. Observe that in the resulting colouring of D, any monochromatic directed cycle C cannot be fully contained in D', nor fully contained in S. In particular there must be an arc of C from $V(D')$ to S, thus to S^+ since S is dipolar, and one from S, thus S^-, to $V(D')$. Now C must intersect both S^- and S^+, which is absurd since all of their colours are different. □

4.3 Building a dipolar set

In this section we give a construction for a dipolar set in an H-free oriented graph D where H is an oriented P_4. We will then show that the dipolar set we construct has bounded dichromatic number if D satisfies the properties given in Scenario 4.2.1.

4.3.1 Closed Tournaments

The simplest case for our construction is when D contains a *strongly connected* tournament J of order $\omega(D)$. Then, we can build a dipolar set consisting of the union of J and a subset of vertices at distance at most two from K.

Let K be a tournament of order $\omega(D)$ contained in D. By definition every vertex $v \in N(K)$ has a non-neighbour in K. Hence, the graph underlying $D[K \cup \{v\}]$ contains an induced P_3. Now, suppose K is strongly connected. Then we get an even more powerful property: Since K is strongly connected there is both an arc from $K \setminus N(v)$ to $N(v) \cap K$ and to $K \setminus N(v)$ from $N(v) \cap K$. This means that $D[K \cup \{v\}]$ contains an induced P_3 starting at v whose last edge is oriented as $\rightarrow$ and an induced P_3 starting at v whose last edge is oriented as $\leftarrow$. This property will give us more power to build specific induced orientations of P_3 in $N[K]$. In particular, this restricts the way vertices at distance at most two interact with the rest of the graph and allows us to exhibit a dipolar set.

To overcome the fact that D may not contain a strongly connected tournament of order $\omega(D)$, we use closed tournaments. By definition of closed tournament every strongly connected oriented graph has a path-minimizing closed tournament. We will base our construction of a dipolar set on some path-minimizing tournament in order to gain some additional structure that we can use to bound the dichromatic number of our dipolar set. In the next subsection we formally give the definition of our dipolar set.

4.3.2 Extending a closed tournament into a dipolar set

In order to build a dipolar set from a closed tournament, we need to make some distinctions between different types of neighbours of a set of vertices. For a set of vertices A and $v \in N(A)$ we say v is a *strong neighbour* of A if v has both an in-neighbour and an out-neighbour in A. Then, the *strong neighbourhood* of A is the set of strong neighbours of A.

Given a closed tournament C, we let X denote the set of strong neighbours of C. The following lemma proves that $N[C \cup X]$ is a dipolar set.

Lemma 4.3.1. *Let H be an orientation of P_4 and D be an H-free oriented graph. Let C be a closed tournament in D and let X denote the strong neighbourhood of C. Then $N[C \cup X]$ is a dipolar set.*

Proof. Let Z denote the neighbours of C that are not strong, and let $Y = N(X) \setminus N[C]$. These sets satisfy $N[C \cup X] = C \cup X \cup Z \cup Y$ and the graph on $N[C \cup X]$ is illustrated in Figure 4.1.

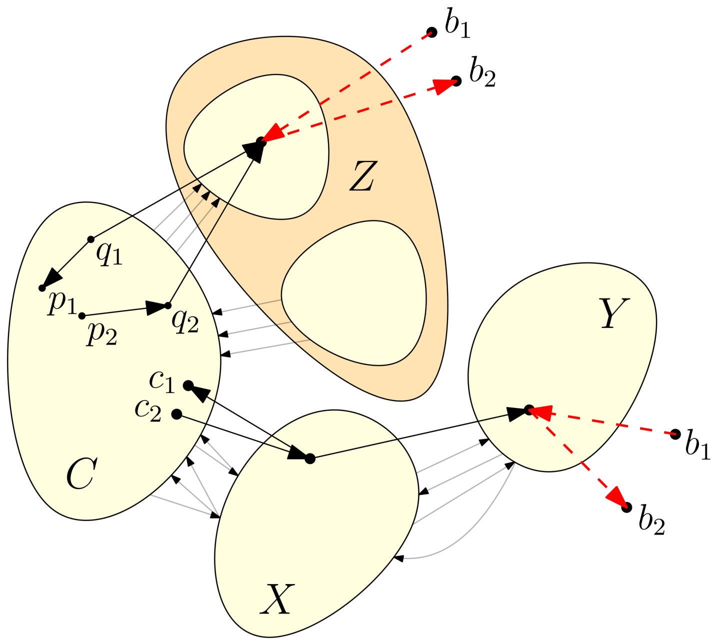


Figure 4.1: An illustration of the extension of a closed tournament C into the dipolar set $N[C \cup X]$. Highlighted in orange, Z consists of neighbours of C that are not strong, i.e., do not have both an in-neighbour and an out-neighbour in C. The set X consists of the strong neighbourhood of C, while set Y contains all neighbours of X not in $N[C]$. Note that arcs between Z and X or Y are not represented here. In Lemma 4.3.1, we prove that if there is some vertex in $N[C \cup X]$ with both an in-neighbour and an out-neighbour in the rest of the oriented graph (drawn in dashed red), then $N[C \cup X] \cup \{b_1, b_2\}$ contains all orientations of P_4 as an induced oriented subgraph.

Then by definition, $N[C \cup X] = N[C] \cup Y$ and the only vertices of $N[C \cup X]$ with neighbours in $V(D) \setminus N[C \cup X]$ are in $Y \cup Z$. Suppose for a contradiction that some $v \in Z \cup Y$ has both an in-neighbour b_1 and an out-neighbour b_2 in $V(D) \setminus (N[C] \cup Y)$. Let us first deal with the case where $v \in Y$.

If $v \in Y$, then $D[C \cup X \cup Y \cup \{b_1, b_2\}]$ contains H. (4.1)

Suppose $v \in Y$. Then, by definition, the following statements all hold:

- There is some $x \in X$ such that x and v are adjacent.
- There are vertices $c_1, c_2 \in C$ where c_1 is an in-neighbour of x and c_2 is an out-neighbour of x.
- b_1, b_2 are not adjacent to any of x, c_1, c_2.

Thus, for some choice of $i, j \in \{1, 2\}$ the set $\{c_i, x, v, b_j\}$ induces a copy of H. (See Figure 4.1.) This proves (4.1).

Since D is an H-free oriented graph, it follows from (4.1) that $v \in Z$. Then by definition of Z, the neighbours of v in C are either all in-neighbours of v or all out-neighbours of v.

There exist arcs $(q_1, p_1), (p_2, q_2) \in E(C)$ such that v is adjacent to q_1, q_2 and non-adjacent to p_1, p_2. (4.2)

It follows from the fact that $\omega(C) = \omega(D)$ that v has some non-neighbour in C. Since C is strongly connected, $N(v) \cap C$ must have both an incoming arc and an outgoing arc from $C \setminus N(v)$. Let p_1, p_2 be vertices of $C \setminus N(v)$ witnessing this fact and let q_1, q_2 their respective neighbours in $N(v) \cap C$. This proves (4.2).

It follows that for some $i, j \in \{1, 2\}$ the graph induced by $\{p_i, q_i, v, b_j\}$ is a copy of H, a contradiction. (See Figure 4.1). □

4.4 First steps towards bounding the dichromatic number of $N[C \cup X]$

For brevity, we will fix the following variables for the remainder of the chapter.

Definition 4.4.1. Let $D, H, \gamma, K, P, C, X, Y$ be defined as follows:

- Let H be an oriented P_4.
- Let D be a digraph satisfying the assumptions of Scenario 4.2.1 with respect to H. Let γ be as in Scenario 4.2.1.
- We choose a tournament K of order $\omega(D)$ and a directed path P that form a path-minimizing tournament C in D.
- Let X be the strong neighbourhood of C and $Y = N(X) \setminus N[C]$.

In the previous section we showed that $N[C \cup X]$ is a dipolar set. Thus, by Lemma 4.2.3 we can prove that all orientations of P_4 are $\vec{\chi}$-bounding by proving that $\vec{\chi}(N[C \cup X])$ is bounded in terms of $\omega(D)$ and γ, the maximum value of $\vec{\chi}(D')$ for any H-free D' with clique number less than $\omega(D)$.

Since $N[C \cup X] \subseteq N[K] \cup V(P) \cup (N(P) \setminus N[K]) \cup Y$, the following inequality holds by definition of the dichromatic number:

$$\vec{\chi}(N[C \cup X]) \leq \vec{\chi}(N[K]) + \vec{\chi}(V(P)) + \vec{\chi}(N(P) \setminus N[K]) + \vec{\chi}(Y) \tag{4.3}$$

We will bound $\vec{\chi}(N[C \cup X])$ by bounding each of the terms on the right-hand side of the inequality. We bound the dichromatic number of $N[K]$ and P in Subsection 4.4.1 and we bound the dichromatic number of Y in Subsection 4.4.2. We are able to use the same techniques for each choice of H when proving these bounds.

As already hinted, bounding $\vec{\chi}(N[C \cup X])$ is non-trivial because we have no control over the cardinality of P. Hence, we cannot obtain a useful bound on $\vec{\chi}(N(P) \setminus N[K])$ by simply applying Observation 4.2.2. In the next section, we will show how to bound $\vec{\chi}(N(P) \setminus N[K])$. We will require separate proofs for $H = \overrightarrow{Q_4}, \overrightarrow{P_4}, \overrightarrow{A_4}$.

4.4.1 Bounding the dichromatic number of $V(P)$ and $N[K]$

We bound the dichromatic number of $V(P)$ and $N[K]$ by an easy observation about "forward-induced" paths. We say a directed path $p_1 \to p_2 \to \cdots \to p_t$ is *forward-induced* if no arc of the form (p_i, p_j) exists where $j > i+1$ and $i, j \in [1, t]$.

Observation 4.4.2. *Let P be a forward-induced directed path in some oriented graph. Then $\vec{\chi}(P) \leq 2$.*

Proof. Let the vertices of P be $p_1 \to p_2 \to \ldots \to p_\ell$, in order. We assign colours to the vertices of P by alternating the colours along P. Suppose there is some monochromatic directed cycle Q in the oriented graph induced by $V(P)$. Then Q contains no arc of P. Hence, Q must contain some arc (p_i, p_j) with $i, j \in [1, \ell]$ and $j > i+1$, contradicting the defintion of forwards-induced. □

Now, we turn to bound the dichromatic number of our dipolar set, $N[C \cup X]$ by bounding $\vec{\chi}(N[K] \cup V(P))$.

Observation 4.4.3. *It holds that*

$$\vec{\chi}(N[K]) \leq \omega \cdot \gamma$$

Moreover,

$$\vec{\chi}(N[C]) \leq \vec{\chi}(N(P) \setminus N[K]) + \omega \cdot \gamma + 2.$$

Proof. Since $|K| = \omega(D) > 1$, we have $N[K] = \bigcup_{x \in K} N(x)$, and hence $\vec{\chi}(N[K]) \leq \omega \cdot \gamma$ by Observation 4.2.2. This proves the first statement. By definition, $N[C] = (N(P) \setminus N[K]) \cup N[K] \cup P$. Since C is path-minimizing, P is forward-induced. Thus, we obtain the second statement by Observation 4.4.2. □

Thus, by inequality (4.3), it only remains to bound the dichromatic number of Y and $N(P) \setminus N[K]$ in order to bound the dichromatic number of our dipolar set $N[C \cup X]$.

4.4.2 Bounding the dichromatic number of Y

In this subsection, we bound the dichromatic number of $Y = N(X) \setminus N[C]$. We first state a more general lemma, which gives the bound on $\vec{\chi}(Y)$ as a direct corollary.

Lemma 4.4.4. *Let H be an oriented P_4 and let D be an H-free oriented graph. Suppose there is a partition of $V(D)$ into sets Q, R, S such that there is no arc between Q and S, every $r \in R$ has both an in-neighbour and an out-neighbour in Q, and every $s \in S$ has a neighbour in R. Let γ be a positive integer such that for every $r \in R$, we have $\vec{\chi}(N(r)) \leq \gamma$. Then $\vec{\chi}(S) \leq 2\gamma$.*

Proof. Note that if an oriented graph D with partition of $V(D)$ into Q, R, S satisfies the conditions of the lemma, for every $S' \subseteq S$ the oriented graph induced by $Q \cup R \cup S'$ also satisfies the conditions of the lemma with partition Q, R, S'. Hence, by Lemma 4.2.3 it is enough to show that for every oriented graph D with partition Q, R, S satisfying the conditions of the lemma and $S \neq \emptyset$, the oriented graph induced by S contains a dipolar set with dichromatic number at most γ.

Let Q, R, S be as in the statement of the lemma and suppose $S \neq \emptyset$. Then, there is some $r \in R$ with a neighbour in S. By assumption, r has an in-neighbour q_1 and an out-neighbour q_2 in Q. Suppose some $s \in N(r) \cap S$ has both an in-neighbour s_i and an out-neighbour s_j in $S \setminus N(r)$. Then there is a copy of H induced by $\{q_i, r, s, s_j\}$ for some choice of $i, j \in \{1, 2\}$, a contradiction. (See

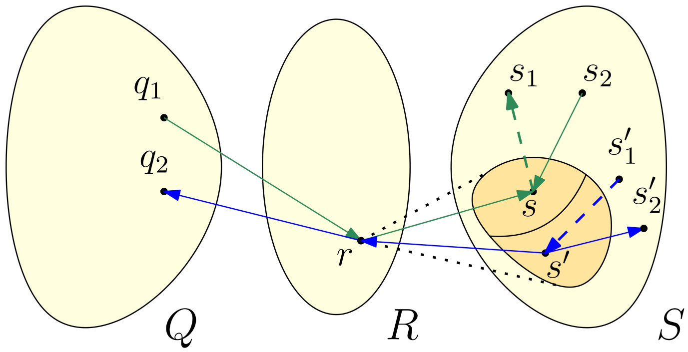


Figure 4.2: Vertex sets Q, R, S, such that no arc lies between Q and S, the vertices in R are all strong neighbours of Q and S is a subset of neighbours of R. We illustrate the case of graphs forbidding $\overrightarrow{P_4}$. Other orientations behave symmetrically. In green, a vertex $r \in R$ is depicted with an out-neighbour $s \in S$. Then if s has an out-neighbour $s_1 \in S \setminus N(r)$ there would be an induced $\overrightarrow{P_4}$, a contradiction. Symmetrically in blue, an in-neighbour $s' \in S$ of r cannot admit an in-neighbour $s_2' \in S \setminus N(r)$.

Figure 4.2.) Hence, $N(r) \cap S$ is a dipolar set. Moreover, $\overrightarrow{\chi}(N(r) \cap S) \leq \overrightarrow{\chi}(N(r)) \leq \gamma$ by definition of γ. □

Corollary 4.4.5. *Under the assumptions of Lemma 4.4.4,*

$$\overrightarrow{\chi}(Y) \leq 2\gamma.$$

Proof. By definition, we may apply Lemma 4.4.4 to the induced subgraph $D[C \cup X \cup Y]$ with $Q \coloneqq C$, $R \coloneqq X$ and $S \coloneqq Y$ (see Figure 4.1). Hence, $\overrightarrow{\chi}(Y) \leq 2\gamma$. □

At this point, we only need to bound $\overrightarrow{\chi}(N(P) \setminus N[K])$ in order to bound the dichromatic number of our dipolar set $N[C \cup X]$ by Corollary 4.4.5. This will be the purpose of the next section.

4.5 Completing the bound on the dichromatic number of our dipolar set

Here, we keep the definitions of D, C, K, P, X, Y and H, γ as fixed in Definition 4.4.1. In this section, we bound the dichromatic number of $N(P) \setminus N[K]$, thus completing the bound on the terms of the right-hand side of inequality (4.3). By Corollary 4.4.5, this will imply that every oriented graph which forbids some orientation of P_4 has a dipolar set of bounded dichromatic number. Thus, by Lemma 4.2.3, this will give us our main result.

By definition of path-minimizing closed tournament, P is a forward-induced directed path. In Subsection 4.5.1, we start by giving some structural properties on properties of the neighbourhood of forwards-induced paths. Then, in Subsection 4.5.2, we show how to use these properties to bound the dichromatic number of the first neighbourhood of C for $\overrightarrow{Q_4}$-free graphs. (Recall, the bound for

$\overrightarrow{Q_4}$-free oriented graphs implies the bound for $\overrightarrow{Q_4}'$-free oriented graphs.) When H is one of the other two orientations, $\overrightarrow{P_4}$ and $\overrightarrow{A_4}$, we required a finer analysis of $N(P)$ in order to bound $\overrightarrow{\chi}(N(P))$. We handle this case in Subsections 4.5.3–4.5.3.

4.5.1 Forbidden arcs amongst neighbours of a forward-induced directed path

We define two partitions of the first neighbourhood of a directed path and show how to forbid some of the arcs between classes of each partition in an H-free oriented graph. For the rest of this chapter we will refer to the vertex set of P as $P = p_1 \to p_2 \to \cdots \to p_\ell$. Many of the results in the following sections actually hold for arbitrary forward-induced paths in D, but we state them for P as this suffices in our proof that orientations of P_4 are $\overrightarrow{\chi}$-bounding.

Definition 4.5.1. For brevity, for any $v \in N(P)$ and $i, j \in [1, \ell]$ we say p_i is the *first* neighbour of v on P if v is adjacent to p_i and non-adjacent to $p_{i'}$ for each $1 \le i' < i \le \ell$. Similarly, p_j is the *last* neighbour of v on P if v is adjacent to p_j and non-adjacent to each $p_{j'}$ for each $1 \le j < j' \le \ell$. We will define two partitions of $N(P)$ according to their first and last neighbours in $V(P)$, respectively.

- For each $i \in [1, \ell]$ we say $v \in N(P)$ is in F_i if p_i is the first neighbour of v on P. This yields partition $(F_1, F_2, \ldots, F_\ell)$, which we call *the partition of* $N(P)$ *by* first attachment *(on* P*).*
- Symmetrically, for each $j \in [1, \ell]$ we say $v \in L_j$ if p_j is the last neighbour of v on P. This yields partition $(L_1, L_2, \ldots, L_\ell)$, which we call the *partition of* $N(P)$ *by last attachment (on* P*).*

For each $i \in [1, \ell]$ we refine each partition by dividing each F_i, L_i into the in-neighbours and out-neighbours of v. We define F_i^+ and L_i^+ to be the sets consisting of all the in-neighbours of p_i in F_i, L_i, respectively. Similarly, we define F_i^- and L_i^- to be the sets consisting of all the out-neighbours of p_i in F_i, L_i, respectively.

Observation 4.5.2. *Let $2 \le i < j \le \ell - 1$. Then the following statements all hold:*

- *If D is $\overrightarrow{Q_4}$-free, there are no arcs from F_j to F_i.*
- *If D is $\overrightarrow{P_4}$-free, there are no arcs from F_i^- to F_j, and no arcs from L_i to L_j^+.*
- *If D is $\overrightarrow{A_4}$-free, there are no arcs from F_i^+ to F_j, and no arcs from L_i to L_j^-.*

Proof. Let $2 \le i < j \le \ell - 1$. We prove each statement individually.

If D is $\overrightarrow{Q_4}$-free, there are no arcs from F_j to F_i. (4.4)

Suppose for some $v \in F_j$ and $w \in F_i$ that $(v, w) \in E(D)$. Then the vertices p_{i-1}, p_i, w, v, induce a P_4 in D with orientation $p_{i-1} \to p_i \to w \leftarrow v$ or orientation $p_{i-1} \to p_i \leftarrow w \leftarrow v$ depending on whether $w \in F_i^+$ or $w \in F_i^-$. In either case we obtain an induced $\overrightarrow{Q_4}$ on p_{i-1}, p_i, w, v. This proves (4.4).

If D is $\overrightarrow{P_4}$-free, there are no arcs from F_i^- to F_j, and no arcs from L_i to L_j^+. (4.5)

Suppose for some $v \in F_i^-$ and $w \in F_j$ that $(v, w) \in E(D)$. Then $p_{i-1} \to p_i \to v \to w$ is an induced $\overrightarrow{P_4}$ (see the dark blue arcs in Figure 4.3). Hence, D is not $\overrightarrow{P_4}$-free. This proves the first part of the statement (4.5). The argument that there are no arcs from L_i to L_j^+ in a $\overrightarrow{P_4}$-free graph is symmetric. This proves (4.5).

If D is $\overrightarrow{A_4}$-free, there are no arcs from F_i^+ to F_j, and no arcs from L_i to L_j^-. (4.6)

By symmetry, it is enough to show that if D is $\overrightarrow{A_4}$-free then there is no arc from F_i^+ to F_j. Suppose for some $v \in F_i^+$ and $w \in F_j$ that $(v, w) \in E(D)$. Then $p_{i-1} \to p_i \leftarrow v \to w$ is an induced $\overrightarrow{A_4}$ in D (see the dark green arcs in Figure 4.3). This proves (4.6). □

In the Subsection 4.5.2 we use Observation 4.5.2 to bound the dichromatic number of $N(P) \setminus N[K]$ in the $\overrightarrow{Q_4}$-free case. In the $\overrightarrow{P_4}$-free case and the $\overrightarrow{A_4}$-free case we need to perform a more careful analysis of $N(P) \setminus N[K]$ in order to bound its dichromatic number because the conditions guaranteed by Observation 4.5.2 are weaker in these two cases. In Subsection 4.5.3, we use Observation 4.5.2 to bound the dichromatic number of the following subsets of $N(P) \setminus N[K]$

$$W^p = \left(F_2^- \cup F_3^- \cup \dots \cup F_{\ell-1}^-\right) \cup \left(L_2^+ \cup L_3^+ \cup \dots \cup L_{\ell-1}^+\right) \quad (4.7)$$

when D is $\overrightarrow{P_4}$-free and

$$W^a = \left(F_2^+ \cup F_3^+ \cup \dots \cup F_{\ell-1}^+\right) \cup \left(L_2^- \cup L_3^- \cup \dots \cup L_{\ell-1}^-\right) \quad (4.8)$$

when D is $\overrightarrow{A_4}$-free. The vertices in $N(P) \setminus (N[K] \cup W^p)$ and $N(P) \setminus (N[K] \cup W^a)$ have restrictions on how they may have neighbours in $V(P)$. We will use this to bound their dichromatic number in Subsections 4.5.3 and 4.5.3, respectively.

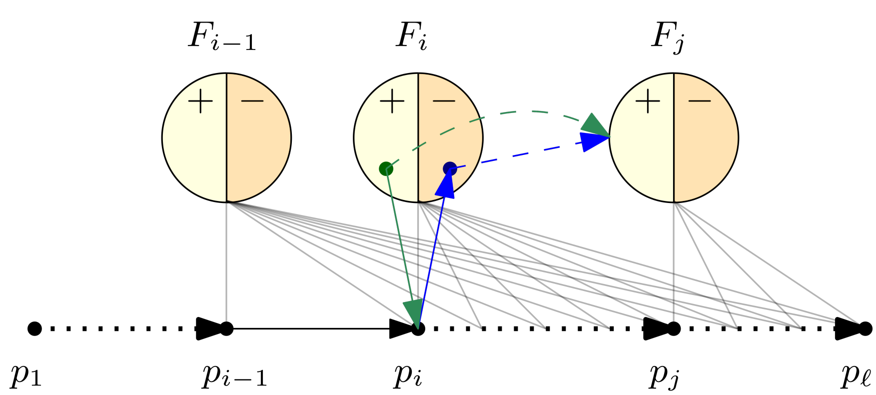


Figure 4.3: A depiction of $P = p_1 \to ... \to p_\ell$ along with the partition $(F_1, ..., F_\ell)$ of $N(P)$ by first attachment on P. Note that the setting is symmetric for the partition of $N(P)$ by the last attachment. Each class of the partition F_i is represented as a circle and further split into F_i^+ in yellow and F_i^- in orange, all possible arcs towards P are drawn in gray. An arc from F_i^- to F_j with $j > i$ would induce a $\overrightarrow{P_4}$ using (p_{i-1}, p_i), as highlighted in blue. An arc from F_i^+ to F_j would induce a $\overrightarrow{A_4}$, represented in green.

4.5.2 The $\overrightarrow{Q_4}$-free case

In this section, we bound the dichromatic number of a path-minimizing closed tournament in D when D is a $\overrightarrow{Q_4}$-free oriented graph satisfying the conditions of Scenario 4.2.1.

Lemma 4.5.3. *If D is $\overrightarrow{Q_4}$-free, $\overrightarrow{\chi}(N(P) \setminus N(\{p_1, p_\ell\})) \leq \gamma$.*

Proof. Assume D is a $\overrightarrow{Q_4}$-free oriented graph. By definition $N(P)\setminus N(\{p_1, p_\ell\}) \subseteq F_2 \cup F_3 \cup \cdots \cup F_{\ell-1}$. By Observation 4.5.2 every directed cycle in $D[F_2 \cup F_3 \cup \cdots \cup F_{\ell-1}]$ is completely contained in $D[F_i]$ for some $i \in [2, \ell-1]$. By definition $F_i \subseteq N(p_i)$ so $\overrightarrow{\chi}(F_i) \leq \gamma$ for each $i \in [1, \ell]$. Hence, we may use the same set of γ colours for each of $F_2, F_3, \ldots, F_{\ell-1}$. Thus, $\overrightarrow{\chi}(N(P) \setminus N(\{p_1, p_\ell\}) \leq \gamma$. □

Lemma 4.5.3 allows us to demonstrate a bound on our dipolar set $N[C \cup X]$ as follows:

Lemma 4.5.4. *Let D be $\overrightarrow{Q_4}$-free. Then D has a dipolar set with dichromatic number at most $(\omega(D) + 3) \cdot \gamma + 2$*

Proof. Recall $C = K \cup P$ is a path-minimizing closed tournament of D and p_1, p_ℓ denote the ends of P. By definition, $p_1, p_\ell \in K$. By Lemma 4.3.1, $N[C \cup X]$ is a dipolar set. Then, by inequality (4.3), bounding the terms on the right-hand side by Observation 4.2.2, Observation 4.4.2, Lemma 4.5.3, and Corollary 4.4.5 we obtain:

$$\overrightarrow{\chi}(N[C \cup X]) \leq +(\omega(D) + 3) \cdot \gamma + 2.$$

□

4.5.3 The $\overrightarrow{P_4}$-free case and the $\overrightarrow{A_4}$-free case

In this subsection, we bound the dichromatic number of our dipolar set in the case where D is $\overrightarrow{P_4}$-free or $\overrightarrow{A_4}$-free.

Bounding $\overrightarrow{\chi}(W^p)$ and $\overrightarrow{\chi}(W^a)$

In this subsection, we bound the dichromatic number of W^p and W^a using Observation 4.5.2 in $\overrightarrow{P_4}$-free and $\overrightarrow{A_4}$-free oriented graphs, respectively.

Lemma 4.5.5. *If D is $\overrightarrow{P_4}$-free, then $\overrightarrow{\chi}(W^p) \leq 2\gamma$. Similarly, if D is $\overrightarrow{A_4}$-free, then $\overrightarrow{\chi}(W^a) \leq 2\gamma$.*

Proof. We begin by proving the first statement. Suppose D is $\overrightarrow{P_4}$-free. Then by Observation 4.5.2, every directed cycle in $D[F_2^- \cup F_3^- \cup \cdots \cup F_{\ell-1}^-]$ is completely contained in $D[F_i^-]$ for some $i \in [2, \ell-1]$. By assumption, $\overrightarrow{\chi}(N(p_i)) \leq \gamma$ for every $p_i \in P$. Hence, we may use the same set of γ colours for each of $F_2^-, F_3^-, \ldots, F_{\ell-1}^-$. So $\overrightarrow{\chi}(F_2^- \cup F_3^- \cup \cdots \cup F_{\ell-1}^-) \leq \gamma$. By symmetry, $\overrightarrow{\chi}(L_2^+ \cup L_2^+ \cup \cdots \cup L_{\ell-1}^+) \leq \gamma$. Therefore, since W^p is the union of these two sets, we obtain $\overrightarrow{\chi}(W^p) \leq 2\gamma$.

The case is symmetric when D is $\overrightarrow{A_4}$-free. The third item of Observation 4.5.2 allows us to use the same set of colours for each of $F_2^+, F_3^+, \ldots, F_{\ell-1}^+$, and the same set of colours for each of $L_2^-, L_3^-, \ldots, L_{\ell-1}^-$. Hence, $\overrightarrow{\chi}(F_2^+ \cup F_3^+ \cup \cdots \cup F_{\ell-1}^+) \leq \gamma$ and $\overrightarrow{\chi}(L_2^- \cup L_3^- \cup \cdots \cup L_{\ell-1}^-) \leq \gamma$. Since W^a is the union of these two sets, $\overrightarrow{\chi}(W^a) \leq 2\gamma$. □

Completing the bound on the dichromatic number of our dipolar set in the $\overrightarrow{P_4}$-free case

Recall, we keep the definitions of D, C, K, P, X, Y and H, γ as fixed in Definition 4.4.1. We will assume $H = \overrightarrow{P_4}$ for the remainder Subsubsection 4.5.3. In this section, we will consider the dichromatic number of the following set of vertices.

Definition 4.5.6. We let

$$R^p = N(P) \setminus (N(\{p_1, p_2, p_\ell\}) \cup W^p)$$

Then, $N(P) \setminus N[K] \subseteq W^p \cup R^p \cup N(p_2)$ and we can bound $\overrightarrow{\chi}(W^p)$ in terms of $\omega(D)$ and γ. Since $\omega(D) = \omega$, it follows that $\omega(N(v)) < \omega$ for each $v \in V(D)$. Hence, $\overrightarrow{\chi}(N(p_2)) \leq \gamma$. Thus, by Lemma 4.3.1 and Corollary 4.4.5, we only need to bound $\overrightarrow{\chi}(R^p)$ in terms of ω and γ in order to demonstrate that D is a dipolar set of bounded dichromatic number.

By definition, W^p is the set of vertices in $N(P) \setminus N(\{p_1, p_\ell\})$ whose first neighbour on P is an in-neighbour or whose last neighbour in $V(P)$ is an out-neighbour. Hence, R^p consists exactly of the vertices in $N(P) \setminus N(\{p_1, p_2, p_\ell\})$ whose first neighbour in $V(P)$ is an out-neighbour and whose last neighbour in $V(P)$ is an in-neighbour.

We will show that since $C = K \cup V(P)$ is a *path-minimizing* closed tournament there is no tournament of order ω in R^p and thus $\overrightarrow{\chi}(R^p) \leq \gamma$. In particular, we will show that for a contradiction, if R^p has a tournament J of order ω, then we can find a directed path P' that is *shorter than* P such that J and P' form a closed tournament. In order to prove this, we will need the following lemma, which will allow us to exhibit a relatively short path between two adjacent vertices in R^p. Note that the following lemma does not use any property of P other than that it is forward-induced.

Lemma 4.5.7. *Let $a, b \in R^p$, if $(b, a) \in E(D)$, there is a directed path from a to b on at most* $\max\{6, \ell - 1\}$ *vertices.*

Proof. Let $\alpha, \beta \in \{1, 2, \ldots, \ell\}$ such that p_α is the *first* neighbour of a in $V(P)$ and p_β is the *last* neighbour of b in $V(P)$. Then, since $a, b \notin \cup_{i=1}^{\ell} F_i^- \cup L_i^+$, the corresponding arcs are $(a, p_\alpha), (p_\beta, b) \in E(D)$. By definition of R^p, we have $3 \leq \alpha, \beta \leq \ell - 1$. Hence, we may assume that $\beta < \alpha$, for otherwise $a \to p_\alpha \to p_{\alpha+1} \to \cdots \to p_\beta \to b$ is a directed path from a to b with at most $\ell - 1$ vertices, as desired.

Since $\beta < \alpha$ the vertices a, b have no common neighbours in $V(P)$. Now, consider the directed path $p_\beta \to b \to a \to p_\alpha$. Since D is $\overrightarrow{P_4}$-free it cannot be induced. Thus, $(p_\beta, p_\alpha) \in E(D)$ or $(p_\alpha, p_\beta) \in E(D)$.

Suppose that $(p_\alpha, p_\beta) \in E(D)$. Then, $v \to p_\alpha \to p_\beta \to b$ is a directed path from a to b of length three, as desired. Hence, we may assume that $(p_\beta, p_\alpha) \in E(D)$.

Since P is a forward-induced directed path and $\beta < \alpha$, it follows that $\alpha = \beta + 1$. Consider the directed path $p_{\beta-1} \to p_\beta \to p_\alpha \to p_{\alpha+1}$. Since D is $\overrightarrow{P_4}$-free it cannot be induced. Therefore, the vertices $p_{\beta-1}$ and p_α, the vertices p_β and $p_{\alpha+1}$, or the vertices $p_{\beta-1}$ and $p_{\alpha+1}$ are adjacent. Furthermore, since P is a shortest path, this means that at least one of $(p_{\alpha+1}, p_{\beta-1})$, $(p_{\alpha+1}, p_\beta)$,$(p_\alpha, p_{\beta-1})$ is an arc of D, see Figure 4.5.3. We consider each case separately:

- Suppose $(p_\alpha, p_{\beta-1})$ is an arc of D. Then $a \to p_\alpha \to p_{\beta-1} \to p_\beta \to b$ is a path of D.
- Suppose $(p_{\alpha+1}, p_\beta)$ is an arc of D. Then $a \to p_\alpha \to p_{\alpha+1} \to p_\beta \to b$ is a path of D.
- Suppose $(p_{\alpha+1}, p_{\beta-1})$ is an arc of D. Then $a \to p_\alpha \to p_{\alpha+1} \to p_{\beta-1} \to p_\beta \to b$ is a path of D.

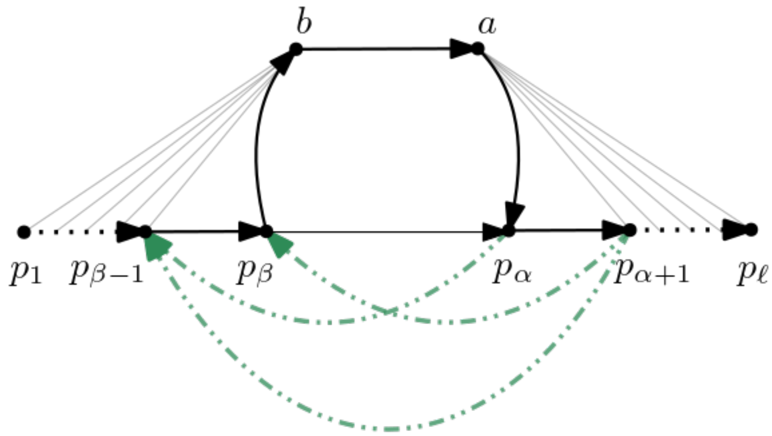


Figure 4.4: On the bottom, P and an arc (b, a) between neighbours of P in R^p. Illustrated here is the case where the last neighbour p_β of b on P appears just before the first neighbour p_α of a, all possible arcs are shown in gray. Then, path $p_{\beta-1}, p_\beta, p_\alpha, p_{\alpha+1}$ cannot induce a $\overrightarrow{P_4}$. Any arc possibly preventing this, shown in dash-dotted green, yields a path from a to b of length at most five.

In every case, the oriented graph induced by $\{a, p_\alpha, p_{\alpha+1}, p_{\beta-1}, p_\beta, b\}$ contains a directed path from a to b on at most six vertices. Since one of the cases must hold, this completes the proof. □

With the last lemma in hand, we are ready to bind the dichromatic number of R^p.

Lemma 4.5.8. *Suppose D is $\overrightarrow{P_4}$-free. Then D contains a dipolar set of dichromatic number at most $(\omega + 6) \cdot \gamma + 2$.*

Proof. Let K be a maximum tournament and P be a directed path in D such that K and P form a path-minimizing closed tournament C in D. Then by Lemma 4.3.1, $N[C \cup X]$ is a dipolar set. We will use Lemma 4.5.7 to bound $\overrightarrow{\chi}(R^p)$. Then we will combine this bound with the results from the previous sections to bound $\overrightarrow{\chi}(N[C \cup X])$.

If P contains at least 7 vertices, then $\overrightarrow{\chi}(R^p) \leq \gamma$. (4.9)

Suppose $\ell \geq 7$. If $\omega(R^p) < \omega(D)$ then by assumption $\overrightarrow{\chi}(R^p) \leq \gamma$, so we may assume that there is an $\omega(D)$-tournament $J \subseteq R$. Since C is a minimum closed tournament and P is non-empty, J is not strongly connected. Hence, there must be exactly one strongly connected component of J that is a sink and exactly one strongly connected component of J that is a source (and they are not equal). Let v be a vertex in the sink component of J and w be a vertex in the source component of J. Therefore, $(w, v) \in E(D)$. Thus, by Lemma 4.5.7 there is a path Q from v to w of length less than that of P. Hence, J, P' form a closed tournament. By definition since K, P were chosen to form a path-minimizing closed tournament P' cannot be shorter than P, a contradiction. This proves (4.9).

$$\overrightarrow{\chi}(N(P) \setminus N[K]) \leq 4\gamma \tag{4.10}$$

Let the vertices of P be $p_1 \to p_2 \to \ldots \to p_\ell$, in order. By definition, $p_1, p_\ell \in K$. If $\ell \leq 6$, then by Observation 4.2.2, $\vec{\chi}(N(P) \setminus N[K]) \leq 4\gamma$, as desired. Hence, we may assume this is not the case.

Let W^p, R^p be defined with respect to P. Then,

$$\vec{\chi}(N(P) \setminus N[K]) \leq \vec{\chi}(W^p) + \vec{\chi}(R^p) + \vec{\chi}(N(p_2)).$$

Hence, by Lemma 4.5.5, Observation 4.2.2, and (4.9) we obtain $\vec{\chi}(N(P) \setminus N[K]) \leq 4\gamma$. This proves (4.10).

Then, by inequality (4.3), bounding the terms on the right-hand side by Observation 4.2.2, Observation 4.4.2, inequality (4.10), and Corollary 4.4.5, we have

$$\vec{\chi}(N[C \cup X]) \leq \vec{\chi}(N(P) \setminus N[K]) + (\omega + 2) \cdot \gamma + 2 \leq (\omega + 6) \cdot \gamma + 2.$$

□

Completing the bound on the dichromatic number of our dipolar set in the $\overrightarrow{A_4}$-free case

In this section, we prove a bound on the remaining vertices of $N(P) \setminus N[K]$ and use the results of the previous sections to show that D contains a dipolar set of bounded dichromatic number in the $\overrightarrow{A_4}$-free case.

Definition 4.5.9. Let P be a shortest path with vertices $p_1 \to p_2 \to \cdots \to p_\ell$, in order. Then

$$R^a = N(P) \setminus (N(\{p_1, p_\ell\}) \cup W^a).$$

Recall $W^a = \cup_{i=2}^{\ell-1} F_i^+ \cup L_i^-$, that is, vertices in $N(P) \setminus N(\{p_1, p_\ell\})$ whose first neighbour on P is an out-neighbour or whose last neighbour in $V(P)$ is an in-neighbour. Hence, R^a consists exactly of the vertices in $N(P) \setminus N(\{p_1, p_\ell\})$ whose first neighbour in $V(P)$ is an in-neighbour and whose last neighbour in $V(P)$ is an out-neighbour.

We bound $\vec{\chi}(R^a)$ using a similar technique to the one we used to bound the dichromatic number of W^a. Recall that we need to bound the dichromatic number of the union of the sets $F_i \backslash W^a$. To this end, we prove that there are no arcs between these sets with indices differing by more than three. We first make the following observation, holding for any shortest directed path.

Observation 4.5.10. *Let D be an $\overrightarrow{A_4}$-free oriented graph. Let $P = p_1 \to p_2 \to \ldots \to p_\ell$ be a shortest directed path from p_1 to p_ℓ in D. Let $i, j \in [1, \ell]$ with $j > i + 3$. Suppose $v, w \in N(P)$ such that $(p_i, v), (w, p_j) \in E(D)$. Then $(v, w) \notin E(D)$.*

Proof. If $(v, w) \in E(D)$ then we may replace the path $p_i \to p_{i+1} \to p_{i+2} \to p_{i+3} \to \cdots \to p_j$ with the path $p_i \to v \to w \to p_j$ in P to obtain a shorter directed path from p_1 to p_ℓ, a contradiction. □

The previous observation allows us to prove that some arcs between vertices in R^a are forbidden.

Observation 4.5.11. *Let D be an $\overrightarrow{A_4}$-free oriented graph. Then, for any two integers in $i, j \in [2, \ell - 1]$ satisfying $i + 2 < j$ there is no arc from a vertex in $F_i \setminus W^a$ to a vertex in $F_j \setminus W^a$.*

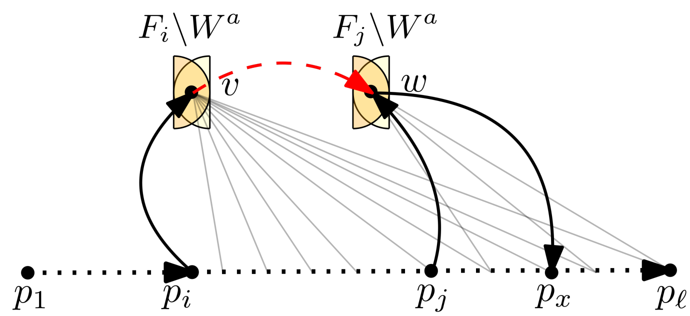


Figure 4.5: A depiction of P, along with $v \in F_i \backslash W^a$ and $w \in F_j \backslash W^a$ of R^a. Since vertex $w \notin W^a$, it must also belong to some L_x^+ with $x > j$, meaning its last neighbour on P is an out-neighbour. Then, since $x > j > i + 2$, an arc (v, w) would yield a shorter path from p_1 to p_ℓ.

Proof. Let i, j be integers $i, j \in [2, \ell - 1]$ satisfying $i + 2 < j$. Suppose $v \in F_i \backslash W^a$ and $w \in F_j \backslash W^a$. By definition of W^a, for every $r \in N(P) \backslash W^a$, the first neighbour of r in $V(P)$ is an in-neighbour of r and the last neighbour of $r \in N(P)$ is an out-neighbour of r. Hence, there is some $x \in [j+1, \ell-1]$ such that p_x is an out-neighbour of w. Thus, $x > i + 3$ and by Observation 4.5.10, $(v, w) \notin E(D)$, see Figure 4.5. □

With the last observation in hand, we are ready to bound $\vec{\chi}(R^a)$, which we do through a similar argument to the case of W^a. The main difference is that here, we use 3 disjoint pallets of γ colours each and choose a colour palette for $F_i \backslash W^a$ according to the index of i modulo 3 (where $i \in [2, \ell - 1]$).

Lemma 4.5.12. *If D is $\overrightarrow{A_4}$-free, then $\vec{\chi}(R^a)) \leq 3\gamma$.*

Proof. By definition $\vec{\chi}(F_i) \leq \gamma$ for each $i \in [1, \ell]$. We fix three disjoint sets S_0, S_1, S_2 of γ colours and dicolour each $F_j \setminus W^a$ for $j \in [2, \ell - 1]$ with set S_i where $i = j \mod 3$. By Observation 4.5.11, $D[F_2 \cup F_3 \cup \cdots \cup F_{\ell-1} \setminus W^a]$ does not contain any monochromatic directed cycle. Hence, $\vec{\chi}(F_2 \cup F_3 \cup \cdots \cup F_{\ell-1} \setminus W^p) \leq 3\gamma$, as desired. □

We combine the previous observation with the results of the previous sections to show that $\overrightarrow{A_4}$-free oriented graphs have a dipolar set of bounded dichromatic number.

Lemma 4.5.13. *Suppose D is an oriented graph satisfying Scenario 4.2.1 with $H = \overrightarrow{A_4}$. Then D contains a dipolar set of dichromatic number at most $(\omega + 7) \cdot \gamma + 2$.*

Proof. By Lemma 4.3.1, $N[C \cup X]$ is a dipolar set. Now, $N(P) \setminus N[K] \subseteq W^a \cup R^a$. Thus, by combining Lemmas 4.5.5 and 4.5.12, we obtain $\vec{\chi}(N(P) \setminus N[K]) \leq 5\gamma$. Then, by combining this bound with inequality (4.3), Observation 4.2.2, Observation 4.4.2, and Corollary 4.4.5 we obtain:

$$\vec{\chi}(N[C \cup X]) \leq (\omega(D) + 7) \cdot \gamma + 2.$$

□

4.6 Orientations of P_4 are $\vec{\chi}$-bounding

In this section, we consider an oriented graph D satisfying Scenario 4.2.1. The previous sections show that D has a dipolar set of bounded dichromatic number. We will use this result and Lemma 4.2.3 to show that oriented graphs not containing some orientation of P_4 are $\vec{\chi}$-bounded.

4.6.1 D contains a dipolar set with bounded dichromatic number

In the previous sections, it is proved that if D does not contain some $H \in \{\overrightarrow{Q_4}, \overrightarrow{P_4}, \overrightarrow{A_4}\}$, then D has a dipolar set of bounded dichromatic number. These results can be summarized in the following lemma.

Lemma 4.6.1. *Let $\omega > 1$ be an integer. Let $H \in \{\overrightarrow{Q_4}, \overrightarrow{P_4}, \overrightarrow{A_4}\}$. Let γ be the maximum value of $\vec{\chi}(D')$ for any H-free oriented graph D' with $\omega(D') < \omega$. Let D be a strongly connected H-free oriented graph with clique number ω. Then,*

- *If $H = \overrightarrow{Q_4}$, then D contains a dipolar set with dichromatic number at most $(\omega + 3) \cdot \gamma + 2$.*
- *If $H = \overrightarrow{P_4}$, then D contains a dipolar set with dichromatic number at most $(\omega + 6) \cdot \gamma + 2$.*
- *If $H = \overrightarrow{A_4}$, then D contains a dipolar set with dichromatic number at most $(\omega + 7) \cdot \gamma + 2$.*

Proof. The result for $H = \overrightarrow{Q_4}, \overrightarrow{P_4}, \overrightarrow{A_4}$ is given in Lemmas 4.5.4, 4.5.8 and 4.5.13, respectively. □

4.6.2 Computing the $\vec{\chi}$-binding function

We will show that an element from the following family of functions is a $\vec{\chi}$-binding function for any class of oriented graphs forbidding a particular orientation of P_4.

Definition 4.6.2. For any integer $c \geq 3$ we let

$$f_c(x) = 2^x (x + c)! + \sum_{i=0}^{x} \frac{2^{i+2}(x + c)!}{(x + c - i)!}$$

for any non-negative integer x.

We will need that f_c satisfies the following recursive properties in order to show that for some c the function f_c is χ-bounding for any class of oriented graph forbidding a particular orientation of P_4.

Observation 4.6.3. *Let $c \geq 3$. Then:*

- *$f_c(x) = 2(x + c)f_c(x - 1) + 4$ for any integer $x \geq 2$, and*
- *$f_c(1) > 1$,*
- *$f_c(x) \leq (x + c)^{x+c+1.5}$ for any integer $x \geq 1$.*

Proof. By definition, since $c \geq 3$, we have $f_c(1) > 2c! > 1$. Hence, the second bullet holds, and we will now prove the first bullet. Let $x \geq 2$ be an integer. Then,

$$f_c(x) = 2(x+c)\left(2^{x-1}(x+c-1)! + \frac{1}{2(x+c)}\sum_{i=0}^{x}\frac{2^{i+2}(x+c)!}{(x+c-i)!}\right).$$

By definition,

$$\sum_{i=0}^{x}\frac{2^{i+2}(x+c)!}{(x+c-i)!} = 2(x+c)\left(\sum_{i=1}^{x}\frac{2^{i+1}(x+c-1)!}{(x+c-i)!}\right) + 4 = 2(x+c)\left(\sum_{i=0}^{x-1}\frac{2^{i+2}(x+c-1)!}{(x+c-i-1)!}\right) + 4.$$

Thus, by combining the previous two equations we obtain $f_c(x) = 2(x+c)f_c(x-1)+4$. This proves the first bullet.

We will complete the proof by showing the third bullet holds. Let $x \geq 1$. By definition,

$$\sum_{i=0}^{x}\frac{2^{i+2}(x+c)!}{(x+c-i)!} = 2^2 + 2^3(x+c) + 2^4(x+c)(x+c-1) + \cdots + 2^{x+2}\frac{(x+c)!}{c!}.$$

Since $c! > 4$ every $i \in [0,x]$ satisfies $\frac{2^{i+2}(x+c)!}{(x+c-i)!} \leq 2^x(x+c)!$. Hence, we obtain

$$\sum_{i=0}^{x}\frac{2^{i+2}(x+c)!}{(x+c-i)!} \leq 2^x(x+1)(x+c)!.$$

$$f_c(x) < 2^x(x+c)! + 2^x(x+1)(x+c)! \leq (x+2)2^x(x+c)!. \tag{4.11}$$

We will use the following well-known equation called *Stirling's Formula* to complete the proof.

Every $n \geq 1$ satisfies

$$\left(n! < \sqrt{2\pi n}\left(\frac{n}{e}\right)^n e^{\frac{1}{12n}}\right). \tag{4.12}$$

Since $c \geq 3$ we obtain the following by combining (4.11) and (4.12).

$$(x+2)2^x(x+c)! < (x+2)2^x\sqrt{2\pi(x+c)}\left(\frac{(x+c)}{e}\right)^{x+c} e^{\frac{1}{12(x+c)}} < (x+c)^{x+c+1.5}.$$

This proves the third bullet. □

4.6.3 $\overrightarrow{\chi}$-boundedness

We are now ready to prove the following more precise version of our main result, Theorem 4.1.2.

Theorem 4.6.4. *Let H be an orientation of P_4, then H-free graphs are $\overrightarrow{\chi}$-bounded. Specifically,*

- *If D is $\overrightarrow{Q_4}$-free or $\overleftarrow{Q_4}$-free, then $\overrightarrow{\chi}(D) \leq (\omega(D)+3)^{\omega(D)+4.5}$,*
- *If D is $\overrightarrow{P_4}$-free, then $\overrightarrow{\chi}(D) \leq (\omega(D)+6)^{\omega(D)+7.5}$, and*

- *If D is $\overrightarrow{A_4}$-free, then* $\overrightarrow{\chi}(D) \leq (\omega(D)+7)^{\omega(D)+8.5}$.

Proof. Let H be an orientation of P_4 Note $\overrightarrow{Q_4}$ can be obtained from $\overrightarrow{Q_4}'$ by reversing the orientation of every edge. Hence, the theorem holds for $\overrightarrow{Q_4}$ if and only if it holds for $\overrightarrow{Q_4'}$. Thus, may assume $H \in \{\overrightarrow{Q_4}, \overrightarrow{P_4}, \overrightarrow{A_4}\}$.

We let $c = 3$ if $H = \overrightarrow{Q_4}$, $c = 6$ if $H = \overrightarrow{P_4}$ and $c = 7$ if $H = \overrightarrow{A_4}$. Then by the third bullet of Observation 4.6.3, it is enough to show that the class of H-free oriented graphs is $\overrightarrow{\chi}$-bounded by f_c.

We have $f_c(1) > 1$ by the first bullet of Observation 4.6.3, so the statement holds for oriented graphs with no arcs. We complete the proof by induction on the clique number. Let $\omega > 1$ be an integer. Suppose every H-free oriented graph D' with clique number less than ω satisfies $\overrightarrow{\chi}(D') \leq f(\omega(D'))$. Let D be an H-free oriented graph with clique number equal to ω. We will show $\overrightarrow{\chi}(D) \leq f_c(\omega)$. We may assume by induction on the number of vertices that D is strongly connected.

By Lemma 4.6.1, D has a dipolar set S with $\overrightarrow{\chi}(S) \leq (\omega+c)\cdot f_c(\omega-1)+2$. Then by Lemma 4.2.3,

$$\overrightarrow{\chi}(D) \leq 2\cdot\overrightarrow{\chi}(S) \leq 2(\omega+c)\cdot f_c(\omega-1)+4.$$

Since $\omega \geq 2$ this implies $\overrightarrow{\chi}(D) \leq f_c(\omega)$ by the second bullet of Observation 4.6.3. This completes the proof. □

4.7 Further directions

Our result is an initial step towards resolving the ACN $\overrightarrow{\chi}$-boundedness conjecture for orientation of paths in general. However, we think we are still far from this result. Our construction of a dipolar set with bounded chromatic number relies heavily on the length of P_4. Similar techniques have been used for triangle-free oriented graphs to obtain the result for $\overrightarrow{P_6}$ [Abo+24]. Still, both our methods and those of the above are ad-hoc, and we do not expect these approaches to generalize for longer paths. The most pressing question is whether an analogue of the Gyárfás path argument can be obtained to show the conjecture for any $\overrightarrow{P_t}$.

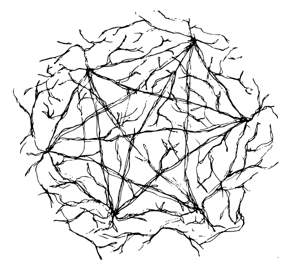
Recall that the classes of $\overrightarrow{Q_4}$-free oriented graphs and $\overrightarrow{Q_4}'$-free oriented graphs were already shown to be χ-bounded in [CSS19]. The χ-binding function f' for these two classes from [CSS19] is defined using recurrence

$$f'(x) \coloneqq 2(3f'(x-1))^5$$

which leads to a double-exponential bound on χ, and cannot guarantee a better bound on $\overrightarrow{\chi}$. In this paper, Theorem 4.1.2 provides an improved $\overrightarrow{\chi}$-binding function when any orientation of P_4 is forbidden. It would interest us to know of any improvements to the $\overrightarrow{\chi}$ function. In particular, we would like to know whether any orientation of P_4 is *polynomially* $\overrightarrow{\chi}$-bounding. In other words, is there some oriented P_4 so that the class of oriented graphs forbidding it has a *polynomial* $\overrightarrow{\chi}$-binding function?

Chapter 5

Brooks' extensions through bivariable degeneracy



This chapter presents our extension of Brooks' theorem to bivariable degeneracy and corresponding linear algorithm, following Section 2.7. It is based on joint results with Daniel Gonçalves and Lucas Picasarri-Arrieta, appearing in [GPR25].

5.1 Introduction

For the usual colouring of graphs, Brooks' Theorem 1.1.7 states that with a palette of $\Delta(G)$ colours, we can always find a colouring of G, except if G is a complete graph or an odd cycle. Such a colouring can be produced in linear time when it exists, see Subsection 2.7.1. For undirected graphs, analogues of Brooks' theorem have been obtained for increasingly general colouring notions (see Theorem 2.7.5, Theorem 2.7.7). This culminated in the following variable degeneracy partition variant called F-colouring [BKT00].

Theorem 5.1.1 (Borodin, Kostochka, Toft '00)**.** *Let G be a connected graph and $F = (f_1, \dots, f_s)$ be a sequence of integer-valued functions such that, for every vertex $v \in V(G)$, $\sum_{i=1}^{s} f_i(v) \geq d(v)$. Then G is F-colourable if and only if (G, F) is not a "hard pair".*

For this notion, only a polynomial-time algorithm is known to produce the associated colouring (or certify its impossibility).

In digraphs, the more general variant presented in Section 2.7 is an analogue of Brooks' theorem for list-dicolouring [HM11], which admits a corresponding linear-time algorithm. In this chapter, we are concerned with the generalization of Theorem 5.1.1 to (di)colourings of digraphs. That is, colouring notions which in fact ask for partitions into classes constrained by "directed variable degeneracy" conditions. This was recently obtained by Bang-Jensen, Schweser, and Stiebitz [BSS22] through f-partitions. They showed a Brooks' analogue, as well as polynomial-time computability of the corresponding colouring.

In this chapter, we define a directed (bi)variable degeneracy colouring called F-dicolouring, generalizing f-partitions, and in turn all previous variants. We show the analogue of Brooks' theorem for this novel notion, and obtain a linear-time algorithm producing the colouring when it exists. As a consequence, this yields the first linear-time algorithms for f-partitions [BSS22], and the F-colouring variant from Theorem 5.1.1. We begin by introducing f-partitions in the following Subsection 5.1.1, before defining F-dicolourings to formally present our results in Subsection 5.1.2.

5.1.1 The previous directed variable degeneracy analogue

To extend the notion of variable degeneracy to digraphs, the authors of [BSS22] consider variable "min-degeneracy", and define the corresponding colouring notion, dubbed *f-partition*. Given a function $h : V(D) \to \mathbb{N}$, a digraph D is h-min-degenerate[1] if for every non-empty subdigraph of D, there exists some $v \in V(D)$ such that $\min(d^-(v), d^+(v)) < h(v)$. Then, letting $f = (f_1, ..., f_s)$, where $f_i : V(D) \to \mathbb{N}$ for any $i \in [s]$, an f-partition of D is a colouring α of $V(D)$ such that vertices coloured i induce an f_i-min-degenerate subgraph for every i. Similar to the case of variable degeneracy, the instances considered are of the form (D, f) such that $\sum_{i=1}^{s} f_i(v) \geq \max(d^-(v), d^+(v))$ for every $v \in V(D)$. Then, the authors define hard pairs in an analogue manner to [BKT00], and show those are exactly the instances that are not f-partionable.

Theorem 5.1.2 (Bang-Jensen, Schweser, Stiebitz [BSS22])**.** *Let D be a connected digraph, let $s \geq 1$ be an integer, and let $f : V(D) \to \mathbb{N}^{\sim}$, be a vector function such that $\sum_{i=1}^{s} f_i(v) \geq \max(d^-(v), d^+(v))$ for all $v \in V(d)$. Then, D is f-partitionable if and only if (D, f) is not a hard pair.*

This notion generalises both dicolouring and list-dicolouring, as well as all variants of undirected colouring presented above. In particular, all the analogues of Brooks' theorem presented until now can be seen as a particular case of Theorem 5.1.2. The authors provide algorithms to produce such colourings or decide of a hard pair, and argue they are polynomial. The time-complexity is not given explicitly, but appears to be at least quadratic.

[1] Actually, in [Bok+04, BSS22] (variable) min-degeneracy is called *(variable) weak degeneracy*, but these names are better suited for our presentation.

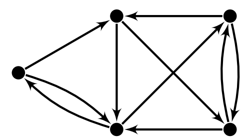


Figure 5.1: A digraph which is strictly-$(2, 0)$-bidegenerate while it is not strictly-$(0, 2)$-bidegenerate.

5.1.2 Our contributions

Our goal is to obtain a linear-time algorithm to construct the colourings guaranteed by the analogues of Brooks' introduced earlier, or decide they are not colourable. The cases for which such an algorithm was missing being Theorem 5.1.1 and Theorem 5.1.2. As the latter is the most general, it is natural to look for a linear algorithm to find the f-partitions considered in [BSS22]. Towards this, we define the more expressive notion of variable bidegeneracy for digraphs, as well as the corresponding F-dicolouring concept. We obtain an analogue of Brooks' theorem for F-dicolouring, as well as a linear-time algorithm that either produces such a colouring, or certifies that there is none. Our strategy does not follow previous approaches in the literature, and the novel point of view given by F-dicolourings is what ultimately leads to linear algorithms for earlier notions as well.

Variable bidegeneracy

The key to our generalisation of Brooks' theorem and corresponding colouring algorithm is the new concept of (variable) *bidegeneracy* for digraphs. Defining degeneracy notions for digraphs can be done in various ways by imposing conditions on the in and out degrees. The first such example in the literature is given by Bokal et al. [Bok+04], who introduced weak degeneracy, which we call min-degeneracy here. Recall a digraph is $(k+1)$-min-degenerate if all its subdigraphs admit a vertex such that the minimum of the in and out degrees is lower than $k+1$. One may also ask for only the in-degree (or only the out-degree) to be lower than $k+1$, which corresponds to the k-in-degeneracy (or k-out-degeneracy) of Bousquet et al. [Bou+24]. We introduce the concept of bidegeneracy for digraphs: for any pair of integers (p, q), a digraph D is *strictly-(p, q)-bidegenerate* if for every subdigraph of D, there exists a vertex v such that either $d^-(v) < p$ or $d^+(v) < q$.

Bidegeneracy allows us to express both min-degeneracy as well as in- or out-degeneracy. Indeed, observe that a digraph D is k-min-degenerate for some $k \in \mathbb{N}$ if and only if it is strictly-(k, k)-degenerate. Then, D is k-in-degenerate (respectively k-out-degenerate) if and only if it is strictly-$(k, 0)$-bidegenerate (respectively strictly-$(0, k)$-bidegenerate). In particular, for a digraph it is equivalent to be acyclic, to be 1-min-degenerate, or to be strictly-$(1, 1)$-bidegenerate. Furthermore, bidegeneracy constraints express a finer range of classes of digraphs, lying strictly between classes defined by other notions. For instance, the digraph illustrated in Figure 5.1.2 is strictly-$(2, 0)$-bidegenerate, thus 2-min-degenerate, yet it is not strictly-$(0, 2)$-bidegenerate. Indeed, bidegeneracy is less symmetric in nature, and importantly, it is not preserved under reversal of all arcs in a given digraph, while min-degeneracy is.

Given any degeneracy notion for (di)graphs, the corresponding variable notion may be defined through a function on the vertices, imposing different restrictions on each vertex of the (di)graph at hand. Notably, the degeneracy of [BSS22] corresponds to the variable version of the degeneracy introduced in [Bok+04]. To the best of our knowledge, there are no known results on the variable in-degeneracy (or variable out-degeneracy). We now define variable bidegeneracy, for a digraph D, through a function $f : V(D) \to \mathbb{N}^2$. We denote the projection of f on the first and second coordinates by f^- and f^+ respectively. Then, D is said to be *strictly-f-bidegenerate* if for every subdigraph H of D, there exists some $v \in V(D)$ such that either $d_H^-(v) < f^-(v)$ or $d_H^+(v) < f^+(v)$. This definition admits an equivalent formulation asking for an elimination ordering of vertices satisfying the above, see Lemma 5.3.2. If $f^-(v) = f^+(v)$ for every $v \in V(D)$, f is said to be symmetric. Again, a digraph is h-min-degenerate if and only if it is strictly-(h, h)-bidegenerate. As a consequence, our notion also generalises all the degeneracy notions used for the analogues of Brooks' theorem presented here.

F-dicolouring

With this in hand, we are ready to define our corresponding notion of dicolourings. Let D be a digraph, s be a positive integer, and $F = (f_1, \ldots, f_s)$ be a sequence of functions $f_i : V(D) \to \mathbb{N}^2$. Then, D is *F-dicolourable* if there exists an s-colouring α of $V(D)$ such that, for every $i \in [s]$, the subdigraph of D induced by the vertices coloured i is strictly-f_i-bidegenerate. Such a colouring α is called an *F-dicolouring*.

Deciding if a digraph is F-dicolourable is clearly NP-hard because it includes a large collection of NP-hard problems for specific values of F. For instance, deciding if a digraph D has dichromatic number at most two, which is NP-hard by [Bok+04], consists exactly in deciding whether D is $((1, 0), (1, 0))$-dicolourable (when considering $(1, 0)$ as a constant function). We thus restrict ourselves to *valid pairs* (D, F), for which D is connected and the following holds:

$$\forall v \in V(D), \quad \sum_{i=1}^{s} f_i^-(v) \geq d^-(v) \quad \text{and} \quad \sum_{i=1}^{s} f_i^+(v) \geq d^+(v). \tag{$\star$}$$

This is a natural condition on the "colour budget" F in terms of the degrees of D, and directly generalises analogues presented earlier. Lowering this budget by only one would allow the expression of hard instances, such as 3-colourability of graphs with maximum degree 4, which is already NP-hard (see [GJ09]).

In Subsection 5.2.1, we define hard pairs (D, F), consisting of a digraph D and a sequence F of functions satisfying Property $(\star)$. The definition is inductive, with the base cases consisting of hard blocks (for which D is biconnected), which may then be glued together by identifying vertices to form other hard pairs. These turn out to be a straightforward adaptation of the hard pairs of [BSS22], and capture other notions of "hard" instances (such as Gallai trees) for the previous analogues of Brooks' theorem. We generalise Brooks' theorem for valid pairs, by showing they are F-dicolourable if and only if they are not hard.

Theorem 5.1.3. *Let (D, F) be a valid pair. Then D is F-dicolourable if and only if (D, F) is not a hard pair (as defined in Subsection 5.2.1). Moreover, there is an algorithm running in time $O(|V(D)| + |A(D)|)$ that decides if (D, F) is a hard pair, and that outputs an F-dicolouring if it is not.*

While the theorem is of independent interest for the new notion of variable bidegeneracy, it is an unifying result allowing for linear algorithms for all the analogues above. Importantly, it yields the first linear-time algorithm for the F-colourability of Theorem 5.1.1 and the f-partitioning of Theorem 5.1.2. Note also that our complexity does not depend on the total number of colours s, such as the algorithm of [Sku06] for Theorem 2.7.5. As the input has size $O(|V(D)|+|A(D)|+s\cdot|V(D)|)$, our algorithm can thus be sublinear in the input size if $(|V(D)|+|A(D)|)$ is asymptotically dominated by $(s \cdot |V(D)|)$. From now on, "linear-time complexity" means linear in the number of vertices and arcs of the considered digraph.

We begin with a few preliminaries and the definition of hard pairs in Section 5.2. Then, Section 5.3 is devoted to the proof of Theorem 5.1.3.

5.2 Preliminaries

Here, a *simple arc* refers to an arc which is not in a digon. Recall the *bidirected graph* obtained from an undirected graph G is denoted by $\overleftrightarrow{G}$, and the *underlying graph* of a digraph D is denoted $\mathrm{UG}(D)$. A digraph is connected, respectively *biconnected* if its underlying graph is (note that K_2 is biconnected). Then, a *block* of D is a maximal biconnected sub(di)graph of D. A cut-vertex of D is a vertex $x \in V(D)$ such that $D - x$ is disconnected. An *end-block* is a block with at most one cut-vertex. We consider any k-colouring as function $\alpha : V(G) \to [k]$, where $[k]$ denotes the set of integers $\{1, .., k\}$. For two elements $p = (p_1, p_2)$ and $q = (q_1, q_2)$ of $\mathbb{N}^2$, we denote by $p \leq q$ the relation $p_1 \leq q_1$ and $p_2 \leq q_2$. We denote by $p < q$ the relation $p \leq q$ and $p \neq q$.

5.2.1 Hard Pairs

We are now ready to define *hard pairs*, which are pairs (D, F) that satisfy Property $(\star)$ tightly, with specific conditions on F, and which admit a certain block tree structure on D. We say that (D, F) is a *hard pair* if one of the following four conditions holds:

(i) D is a biconnected digraph and there exists $i \in [s]$ such that, for every vertex $v \in V$, $f_i(v) = (d^-(v), d^+(v))$ and $f_k(v) = (0, 0)$ when $k \neq i$.

We refer to such a hard pair as a *monochromatic hard pair*.

(ii) D is a bidirected odd cycle and the functions $f_1, \dots, f_s$ are all constant equal to $(0, 0)$ except exactly two that are constant equal to $(1, 1)$.

We refer to such a hard pair as a *bicycle hard pair*.

(iii) D is a bidirected complete graph, the functions $f_1, \dots, f_s$ are all constant and symmetric, and for every vertex v we have $\sum_{i=1}^{s} f_i^+(v) = |V(D)| - 1$.

We refer to such a hard pair as a *complete hard pair*.

(iv) (D, F) is obtained from two hard pairs (D^1, F^1) and (D^2, F^2) by identifying two vertices $x_1 \in V(D^1)$ and $x_2 \in V(D^2)$ into a new vertex $x \in V(D)$, such that for every vertex $v \in V(D)$ we have:

$$f_k(v) = \begin{cases} f_k^1(v) & \text{if } v \in V(D_1) \setminus \{x_1\} \\ f_k^2(v) & \text{if } v \in V(D_2) \setminus \{x_2\} \\ f_k^1(v) + f_k^2(v) & \text{if } v = x. \end{cases}$$

where $F^1 = (f_1^1, \dots, f_s^1)$, $F^2 = (f_1^2, \dots, f_s^2)$.

We refer to such a hard pair as a *join hard pair*.

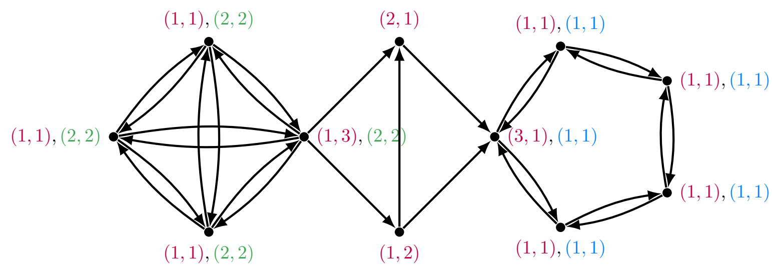


Figure 5.2: An example of a hard pair (D, F) where F is a sequence of three functions f_1, f_2, f_3 the values of which are respectively represented in red, green, and blue. For the sake of readability, the value of f_i is missing when it is equal to $(0, 0)$.

See Figure 5.2 for an illustration of a hard pair.

In the following, we give a short proof that if (D, F) is a hard pair then D is not F-dicolourable.

5.2.2 Hard pairs are not dicolourable

Lemma 5.2.1. *Let (D, F) be a hard pair, then D is not F-dicolourable.*

Proof. Let $(D, F = (f_1, \dots, f_s))$ be a hard pair. Assume for a contradiction that it admits an F-dicolouring α with colour classes $V_1, \dots, V_s$. We distinguish four cases, depending on the kind of hard pair (D, F) is.

(*i*) If (D, F) is a monochromatic hard pair, let $i \in [s]$ be such that, for every vertex $v \in V$, $f_i(v) = (d^-(v), d^+(v))$ and $f_k(v) = (0, 0)$ when $k \neq i$. Then, for every $k \neq i$, V_k must be empty, for otherwise $D[V_k]$ is not strictly-f_k-bidegenerate. Therefore, we have $V = V_i$ and D must be strictly-f_i-bidegenerate. Hence D contains a vertex v such that $d^-(v) < f_i^-(v)$ or $d^+(v) < f_i^+(v)$, a contradiction.

(*ii*) If (D, F) is a bicycle hard pair, let $i, j \in [s]$ be distinct integers such that, for every vertex $v \in V$, $f_i(v) = f_j(v) = (1, 1)$ and $f_k(v) = (0, 0)$ when $k \notin \{i, j\}$. Again, V_k must be empty when $k \notin \{i, j\}$, so (V_i, V_j) partitions V. Since D is a bidirected odd cycle, it is not bipartite, so $D[V_i]$ or $D[V_j]$ contains a digon between vertices u and v. Assume that $D[V_i]$ does, then $H = D[\{u, v\}]$ must contain a vertex x such that $d_H^-(x) < f_i^-(x)$ or $d_H^+(x) < f_i^+(x)$, a contradiction since $f_i(x) = (d_H^-(x), d_H^+(x)) = (1, 1)$ for every $x \in \{u, v\}$.

(*iii*) If (D, F) is a complete hard pair, then for each $i \in [s]$, since $D[V_i]$ is strictly-f_i-bidegenerate, we have $|V_i| \leq f_i^-(v)$ where v is any vertex (recall that the functions $f_1, \dots, f_s$ are constant and symmetric). Then $\sum_{i=1}^s |V_i| \leq \sum_{i=1}^s f_i^-(v) = |V(D)| - 1$, a contradiction since $(V_1, \dots, V_s)$ partitions V.

(*iv*) Finally, if (D, F) is a join hard pair, we proceed by structural induction on the joins, with the three base cases corresponding to the previous items. Now, (D, F) is obtained from two hard pairs (D^1, F^1) and (D^2, F^2) by identifying two vertices $x_1 \in V(D^1)$ and $x_2 \in V(D^2)$ into a new vertex $x \in V(D)$, such that for every vertex $v \in V(D)$ we have:

$$f_k(v) = \begin{cases} f_k^1(v) & \text{if } v \in V(D_1) \setminus \{x_1\} \\ f_k^2(v) & \text{if } v \in V(D_2) \setminus \{x_2\} \\ f_k^1(v) + f_k^2(v) & \text{if } v = x. \end{cases}$$

where $F^1 = (f_1^1, \dots, f_s^1)$, $F^2 = (f_1^2, \dots, f_s^2)$.

By induction, we may assume that D^1 is not F^1-dicolourable and D^2 is not F^2-dicolourable. Let α^1 and α^2 be the following s-colourings of D^1 and D^2.

$$\alpha^1(v) = \begin{cases} \alpha(v) \text{ if } v \neq x_1 \\ \alpha(x) \text{ otherwise.} \end{cases} \quad \text{and} \quad \alpha^2(v) = \begin{cases} \alpha(v) \text{ if } v \neq x_2 \\ \alpha(x) \text{ otherwise.} \end{cases}$$

We denote respectively by $U_1, \dots, U_s$ and $W_1, \dots, W_s$ the colour classes of α^1 and α^2. Since D^1 is not F^1-dicolourable, there exist $i \in [s]$ and a subdigraph H^1 of $D^1[U_i]$ such that every vertex $u \in V(H^1)$ satisfies $d^-_{H^1}(u) \geq {f_i^1}^-(u)$ and $d^+_{H^1}(u) \geq {f_i^1}^+(u)$. We claim that $x_1 \in V(H^1)$. Assume not, then H^1 is a subdigraph of $D[V_i]$ and every vertex u in H^1 satisfies $f_i(u) = f_i^1(u)$, and so it satisfies $d^-_{H^1}(u) \geq f_i^-(u)$ and $d^+_{H^1}(u) \geq f_i^+(u)$. This is a contradiction to $D[V_i]$ being strictly-f_i-bidegenerate.

Hence, we know that $x_1 \in V(H^1)$. Swapping the roles of D^1 and D^2, there exists an index $j \in [s]$ and a subdigraph H^2 of $D^2[W_j]$ such that every vertex $w \in V(H^2)$ satisfies $d^-_{H^2}(w) \geq {f_j^2}^-(w)$ and $d^+_{H^2}(w) \geq {f_j^1}^+(w)$. Analogously, we must have $x_2 \in V(H^2)$.

By the definitions of α^1 and α^2, we have $i = j$ and $x \in V_i$. Let H be the subdigraph of $D[V_i]$ obtained from H_1 and H_2 by identifying x_1 and x_2 into x. For every vertex $u \in V(H) \cap V(H^1)$, we have $d^-_H(u) = d^-_{H^1}(u) \geq f_i^-(u)$ and $d^+_H(u) = d^+_{H^1}(u) \geq f_i^+(u)$. Analogously, for every vertex $w \in V(H) \cap V(H^2)$, we have $d^-_H(w) = d^-_{H^2}(w) \geq {f_i^2}^-(w)$ and $d^+_H(w) = d^+_{H^2}(w) \geq {f_i^1}^+(w)$. Finally, we have

$$\begin{aligned} & d^-_H(x) = d^-_{H^1}(x_1) + d^-_{H^2}(x_2) \geq {f_i^1}^-(x_1) + {f_i^2}^-(x_2) = f_i^-(x) \\ \text{and} \quad & d^+_H(x) = d^+_{H^1}(x_1) + d^+_{H^2}(x_2) \geq {f_i^1}^+(x_1) + {f_i^2}^+(x_2) = f_i^+(x). \end{aligned}$$

This is a contradiction to $D[V_i]$ being strictly-f_i-bidegenerate.

□

5.3 Dicolouring non-hard pairs in linear time

In this section, we prove Theorem 5.1.3 by describing the corresponding algorithm, proving its correctness, and its time complexity. Our algorithm starts by testing whether (D, F) is a hard pair. If it is not, we pipeline various algorithms reducing the initial instance, while partially colouring the vertices pruned along the way, until a solution is found.

We first discuss the data structure encoding the input in Subsection 5.3.1. We give some technical definitions used in the proof and some preliminary remarks in Subsection 5.3.2. In the following subsections we describe different reduction steps unfolding in our algorithm. We consider valid pairs for which the constraints are "loose" in Subsection 5.3.4, based on a simple greedy algorithm presented in Subsection 5.3.3. Then, we consider "tight" valid pairs. In Subsection 5.3.5, we show how to reduce a tight and valid pair (D, F) into another one (D', F') for which D' is biconnected and smaller than D. At this point, we may test the hardness of (D', F') to decide whether (D, F) is hard. Otherwise, the remainder of the section consists in exhibiting an F'-dicolouring of D', yielding an F-dicolouring of D. In Subsection 5.3.6, we reduce the instance to one involving at most two colours, and which is still biconnected. Then, we show how to solve biconnected instances with two colours in Subsection 5.3.7 and in Subsection 5.3.8, through the use of ear-decompositions. We conclude with the proof of Theorem 5.1.3 in Subsection 5.3.9.

5.3.1 Data structures

We need appropriate data structures to process the entry pair (D, F). The following structures are standard, but let us list their properties. The digraph D is encoded in space $O(|V(D)| + |A(D)|)$ with a data structure allowing, for every vertex v,

- to access the values $|V(D)|$, $|A(D)|$, $d^-(v)$, $d^+(v)$, and $|N^-(v) \cap N^+(v)|$ in $O(1)$ time,
- to enumerate the vertices of the sets $V(D)$, $N^-(v)$, $N^+(v)$ in $O(1)$ time per vertex,
- to delete a vertex v (and update all the related values and sets) in $O(d^-(v) + d^+(v))$ time, and
- to compute a spanning tree rooted at a specified root in $O(|V(D)| + |A(D)|)$ time.

The functions $F = (f_i)_{i \in [s]}$ are encoded in $O(s \cdot |V(D)|)$ space in a data structure allowing, for every vertex v,

- to read or modify $f_i^-(v)$ and $f_i^+(v)$ in $O(1)$ time.
- to enumerate (only) the colours i such that $f_i(v) \neq (0, 0)$, in $O(1)$ time per such colour.

This data structure is simply an $s \times |V(D)|$ table, dynamically maintained, with pointers linking the cells (i, v) and (j, v) if $f_i(v) \neq (0, 0)$, $f_j(v) \neq (0, 0)$, and $f_k(v) = (0, 0)$ for every k such that $i < k < j$.

The output is a vertex colouring which we build along the different steps of our algorithm, and is simply encoded in a table.

5.3.2 Preliminaries

Let $(D, F = (f_1, \ldots, f_s))$ be a (non-necessarily valid) pair, let $X \subseteq V(D)$ be a subset of vertices of D, and $\alpha : X \to [s]$ be a partial s-colouring of D. Let $D' = D - X$ and $F' = (f'_1, \ldots, f'_s)$ be defined as follows:

$$\begin{aligned} f_i'^-(u) &= \max(0, f_i^-(u) - |\alpha^{-1}(i) \cap N^-(u)|) \\ \text{and} \quad f_i'^+(u) &= \max(0, f_i^+(u) - |\alpha^{-1}(i) \cap N^+(u)|) \end{aligned}$$

We call (D', F') the pair *reduced* from (D, F) by α.

Lemma 5.3.1. *Consider a (non-necessarily valid) pair (D, F) and an F-dicolouring α of $D[X]$, for a subset $X \subseteq V(D)$. Let (D', F') be the pair reduced from (D, F) by α. Then, combining α with any F'-dicolouring of D' yields an F-dicolouring of D.*

Furthermore, there is an algorithm that given a pair (D, F), and a colouring α of some set $X \subseteq V(D)$, outputs the reduced pair (D', F') in $O(\sum_{v \in X} d(v))$ time.

Proof. Let β be any F'-dicolouring of D'. We will show that the combination γ of α and β is necessarily an F-dicolouring of D. We formally have

$$\gamma(v) = \begin{cases} \alpha(v) & \text{if } v \in X \\ \beta(v) & \text{otherwise.} \end{cases}$$

Letting $i \in [s]$ be any colour, we will show that $D[V_i]$ is strictly-f_i-bidegenerate, where $V_i = \gamma^{-1}(i)$, which implies the first part of the statement. To this purpose, let H be any subdigraph of $D[V_i]$, we will show that H contains a vertex v satisfying $d^-_H(v) < f^-_i(v)$ or $d^+_H(v) < f^+_i(v)$. Observe first that, if $V(H) \subseteq X$, the existence of v is guaranteed since α is an F-dicolouring of $D[X]$. Henceforth assume that $V(H) \setminus X \neq \emptyset$, and let H' be $H - X$. Since $V(H') \subseteq \beta^{-1}(i)$, by hypothesis on β, the digraph H' is strictly-f'_i-bidegenerate. Hence there must be a vertex $v \in V(H')$ such that $d^-_{H'}(v) < {f'_i}^-(v)$ or $d^+_{H'}(v) < {f'_i}^+(v)$. If $d^-_{H'}(v) < {f'_i}^-(v)$, we obtain

$$d^-_H(v) \leq d^-_{H'}(v) + |\alpha^{-1}(i) \cap N^-(v)| < {f'_i}^-(v) + |\alpha^{-1}(i) \cap N^-(v)| = f^-_i(v).$$

Symmetrically, $d^+_{H'}(v) < {f'_i}^+(v)$ implies $d^+_H(v) < f^+_i(v)$, showing that the combined colouring is indeed an F-dicolouring.

The algorithm is elementary. It consists on a loop over vertices $v \in X$, updating the table storing the colouring with colour $\alpha(v)$ for v, deleting v from D, and visiting every neighbour u of v in order to update its adjacency list (that is removing v from $N^-(u)$ and/or $N^+(u)$), and its colour budget $f_{\alpha(v)}(u)$. This is clearly linear in $\sum_{v \in X} d(v)$. □

Let (D, F) be a valid pair. Let X be a subset of vertices of D, $\alpha : X \to [s]$ be a partial colouring of D, and (D', F') be the pair reduced from (D, F) by α. We say that the colouring α of $D[X]$ is *safe* if each of the following holds:

- α is an F-dicolouring of $D[X]$,
- D' is connected, and
- (D, F) is a hard pair if and only if (D', F') is a hard pair.

For the particular case $X = \{v\}$, we thus say that colouring v with c is *safe* if $f_c(v) \neq (0, 0)$, $D - x$ is connected, and the reduced pair (D', F') is a hard pair if and only if (D, F) is.

The following lemma tells us how variable bidegeneracy relates to vertex orderings.

Lemma 5.3.2. *Given a digraph $D = (V, A)$, and a function $f : V \to \mathbb{N}^2$, D is strictly-f-bidegenerate if and only if there exists an ordering $v_1, \ldots, v_n$ of the vertices of D such that*

$$f^-(v_i) > |N^-(v_i) \cap \{v_j \mid j \leq i\}| \quad \text{or} \quad f^+(v_i) > |N^+(v_i) \cap \{v_j \mid j \leq i\}|$$

Proof. ($\Longrightarrow$) We proceed by induction on the number of vertices n. This implication clearly holds for $n = 1$. Assume now that $n \geq 2$. Let v_n be a vertex of D such that $f^-(v_n) > d^-(v_n)$ or $f^+(v_n) > d^+(v_n)$, which exists by strict bidegeneracy of D. Since $d^-(v_n) \geq |N^-(v_n) \cap S|$ and $d^+(v_n) \geq |N^+(v_n) \cap S|$ for any set $S \subseteq V$, the property holds for v_n in any ordering such that v_n is the last vertex. By induction there is an ordering $v_1, \dots, v_{n-1}$ of the vertices of $D' = D - v_n$ such that for every vertex v_i, with $1 \leq i \leq n-1$, we have

$$f^-(v_i) > |N^-_{D'}(v_i) \cap \{v_j \mid j \leq i\}| \quad \text{or} \quad f^+(v_i) > |N^+_{D'}(v_i) \cap \{v_j \mid j \leq i\}|$$

As $N^-_{D'}(v) \cap \{v_j \mid j \leq i\} = N^-_D(v) \cap \{v_j \mid j \leq i\}$ and $N^+_{D'}(v) \cap \{v_j \mid j \leq i\} = N^+_D(v) \cap \{v_j \mid j \leq i\}$, the property still holds in D, concluding this direction.

($\Longleftarrow$) For any subdigraph $H = (V_H, A_H)$ of D let v_k be the vertex of V_H with the largest index k. Since $N^-_H(v_k) = N^-_D(v_k) \cap V_H \subseteq N^-_D(v_k) \cap \{v_j \mid j \leq k\}$, and $N^+_H(v_k) = N^+_D(v_k) \cap V_H \subseteq N^+_D(v_k) \cap \{v_j \mid j \leq k\}$, we have:

$$f^-(v_k) > |N^-_D(v) \cap \{v_j \mid j \leq k\}| \geq d^-_H(v_k) \quad \text{or} \quad f^+(v_i) > |N^+_D(v) \cap \{v_j \mid j \leq i\}| \geq d^+_H(v_k).$$

Therefore, H is strictly-f-bidegenerate. □

5.3.3 Greedy algorithm

In this subsection we consider an algorithm that greedily colours D, partially or entirely. We are going to give conditions ensuring that this approach succeeds in providing a (partial) F-dicolouring.

Lemma 5.3.3. *Given a (non-necessarily valid) pair (D, F) and an ordered list of vertices $v_1, \dots, v_\ell$ of $V(D)$, there exists an algorithm, running in time $O\left(\sum_{i=1}^{\ell} d(v_i)\right)$, that colours the vertices $v_1, \dots, v_\ell$ such that:*

- *If F satisfies the following for every $v_i \in \{v_1, \dots, v_\ell\}$,*

$$\sum_{c=1}^{s} f^-_c(v_i) > |N^-(v_i) \cap \{v_j \mid j \leq i\}| \quad \text{or} \quad \sum_{c=1}^{s} f^+_c(v_i) > |N^+(v_i) \cap \{v_j \mid j \leq i\}| \qquad (\star\star)$$

  then the algorithm produces an F-dicolouring of $D[v_1, ..., v_\ell]$, and computes the reduced pair (D', F').

- *If $(\star\star)$ holds, and moreover (D, F) is valid and $D - \{v_1, \dots, v_\ell\}$ is connected, then the reduced pair (D', F') is valid.*

Note that as $\sum_{v \in D} d(v) = 2|A(D)|$ the time complexity here is $O(|V(D)| + |A(D)|)$, even when the whole digraph is coloured.

Proof. The algorithm simply consists in considering the vertices $v_1, \dots, v_\ell$ in this order and, if possible, to colour v_i with a colour c such that $f^-_c(v_i) > |\{u \in N^-(v_i) \cap \{v_1, \dots, v_i\} \mid u \text{ is coloured } c\}|$, or such that $f^+_c(v_i) > |\{u \in N^+(v_i) \cap \{v_1, \dots, v_i\} \mid u \text{ is coloured } c\}|$.

The complexity of the algorithm holds because, it actually consists in a loop over vertices $v_1, \dots, v_\ell$, where 1) it looks for a colour c such that $f_c(v_i) \neq (0, 0)$ (where F is updated after each vertex colouring), and if it finds such a colour, 2) colours v_i and updates (D, F) into the corresponding reduced pair (D', F'). Step 1) is done in constant time (see Subsection 5.3.1), and step 2) is done in $O(d(v_i))$ time (by Lemma 5.3.1). Hence, the complexity clearly follows. Lemma 5.3.2, applied to $D[\{v_1, ..., v_\ell\}]$, implies that if the algorithm succeeds, the obtained colouring is an F-dicolouring of $D[\{v_1, ..., v_\ell\}]$.

To show the second statement, let us show the following invariant of the algorithm:

($\clubsuit$) *At the beginning of the i^{th} iteration of the main loop, for any vertex v_k with $i \leq k \leq \ell$, at least one of the following occurs:*

- *its number of uncoloured in-neighbours with index at most k is less than $\sum_{c=1}^{s} f_c^-(v_k)$,*
- *its number of uncoloured out-neighbours with index at most k is less than $\sum_{c=1}^{s} f_c^+(v_k)$,*

where the sums are made on the updated functions f_c.

Note that, by ($\star\star$), ($\clubsuit$) holds for the first iteration. Let us show that if ($\clubsuit$) holds at the beginning of the i^{th} iteration for some vertex v_k with $i < k$, then it still holds at the beginning of the $i + 1^{\text{th}}$ iteration. Indeed, during the i^{th} iteration, if the sum $\sum_{c=1}^{s} f_c^-(v_k)$ decreases, it decreases by exactly one, and in that case we have $v_i \in N^-(v_k)$. Hence, the number of uncoloured in-neighbours of v_k with index at most k also decreases by one, and ($\clubsuit$) still holds. The same holds for the out-neighbourhood of v_i.

By ($\clubsuit$), at the beginning of the i^{th} iteration, we have $\sum_{i=1}^{s} f_i(v) \neq (0, 0)$. Hence, there is a colour c such that $f_c(v_i) \neq (0, 0)$, so there is always a colour (*e.g.* c) available for colouring v_i, and the algorithm thus succeeds in colouring the whole digraph $D[v_1, \ldots, v_\ell]$.

Finally for the last statement, if ($\star\star$) holds, then the algorithm succeeds in producing an F-dicolouring. Furthermore, if (D, F) is valid, then for any vertex $u \in V(D) \setminus \{v_1, \ldots, v_\ell\}$, we have $d_D^-(u) \leq \sum_{c=1}^{s} f_c^-(v_i)$ and $d_D^+(u) \leq \sum_{c=1}^{s} f_c^+(v_i)$. For each of these inequalities, and at each iteration of the main loop, the left-hand side decreases if and only if the right-hand side does, in the reduced pair (D', F'). Hence, ($\star$) holds in (D', F'), and since D' is connected, the pair (D', F') is valid. □

5.3.4 Solving loose instances

If the input digraph D as a whole is strictly-$\tilde{f}$-bidegenerate for some $\tilde{f}$, it may be easy to produce an F-dicolouring, under some conditions on $\tilde{f}$. Note that, in what follows, we do not ask for (D, F) to be a valid pair, so D may be disconnected and vertices v do not necessarily satisfy ($\star$).

Lemma 5.3.4. *Let $D = (V, A)$ be a digraph, $F = (f_1, \ldots, f_s)$ be a sequence of functions $f_i : V \to \mathbb{N}^2$, and $\tilde{f} = \sum_{i=1}^{s} f_i$. If D is strictly-$\tilde{f}$-bidegenerate, then D is F-dicolourable and an F-dicolouring can be computed in linear time.*

Proof. Let $(D = (V, A), F = (f_1, \ldots, f_s))$ be such a pair. By Lemma 5.3.2, there exists an ordering $\sigma = v_1, \ldots, v_n$ of $V(D)$ such that, for every $i \in [s]$, v_i satisfies $d_{D_i}^+(v_i) < \tilde{f}^+(v_i)$ or $d_{D_i}^-(v_i) < \tilde{f}^-(v_i)$, where D_i is the subdigraph of D induced by $\{v_1, \ldots, v_i\}$.

We prove that such an ordering σ can be computed in linear time. Our proof follows a classical linear-time algorithm for computing the usual degeneracy of a graph, but we give the proof for completeness. We create two tables *out_gap* and *in_gap*, both of size n and indexed by V. We iterate once over the vertices of V and, for every vertex v, we set *out_gap*(v) to $\tilde{f}^+(v) - d^+(v)$ and

$in\_gap(v)$ to $\tilde{f}^-(v) - d^-(v)$. We also store in a set S all vertices v for which $out\_gap(v) \geq 1$ or $in\_gap(v) \geq 1$.

Then for i going from n to 1, we choose a vertex u in S that has not been treated before, we set v_i to u, and for every in-neighbour (respectively out-neighbour) w of u that has not been treated before, we increase $in\_gap(w)$ (respectively $out\_gap(w)$) by one. If $out\_gap(w) \geq 1$ or $in\_gap(w) \geq 1$, we add w to S. We then remember (using a boolean table for instance) that u has been treated.

Following this linear-time algorithm, at the beginning of each step $i \in [n]$, for every non-treated vertex u, we have $in\_gap(u) = \tilde{f}^-(u) - d^-_{D_i}(u)$ and $out\_gap(u) = \tilde{f}^+(u) - d^+_{D_i}(u)$. We also maintain that S contains all the vertices u satisfying $in\_gap(u) \geq 1$ or $out\_gap(u) \geq 1$. Since D is strictly-$\tilde{f}$-bidegenerate, at step i, there exists a non-treated vertex $u \in S$.

We now admit that such an ordering $\sigma = (v_1, \dots, v_n)$ has been computed, and we greedily colour the vertices from v_1 to v_n. The result follows from Lemma 5.3.3 and the fact that $\tilde{f} = \sum_{i=1}^{s} f_i$. □

A valid pair is *tight*, if for every vertex the two inequalities of ($\star$) are equalities. The following particular case of Lemma 5.3.4 treats the case of instances that are non-tight, which we also refer to as *loose*.

Lemma 5.3.5. *Let (D, F) be a valid pair that is loose. Then D is F-dicolourable and an F-dicolouring can be computed in linear time.*

Proof. Let $(D = (V, A), F = (f_1, \dots, f_s))$ be such a pair and let $\tilde{f} : V \to \mathbb{N}^2$ be $\sum_{i=1}^{s} f_i$. It is sufficient to prove that D is necessarily strictly-$\tilde{f}$-bidegenerate, so the result follows from Lemma 5.3.4.

Assume for a contradiction that D is not strictly-$\tilde{f}$-bidegenerate, so by definition there exists an induced subdigraph H of D such that, for every vertex $v \in V(H)$, $d^+_H(v) \geq \tilde{f}^+(v)$ and $d^-_H(v) \geq \tilde{f}^-(v)$. By the definition of $\tilde{f}$, and because (D, F) is a valid pair, we thus have $d^+_H(v) = \sum_{i=1}^{s} f_i^+(v)$ and $d^-_H(v) = \tilde{f}^-(v)$. We directly deduce that $H \neq D$ as (D, F) is loose.

Since D is connected (as (D, F) is a valid pair) and because $H \neq D$, there exists in D an arc between vertices u and v such that $u \in V(H)$ and $v \in V(D) \setminus V(H)$. Depending on the orientation of this arc, we have $d^+_H(u) < d^+_D(u) = \sum_{i=1}^{s} f_i^+(u)$ or $d^-_H(u) < d^-_D(u) = \sum_{i=1}^{s} f_i^-(u)$, a contradiction. □

5.3.5 Reducing to a block and detecting hard pairs

We have just shown how to solve loose instances. In this subsection, we thus consider a tight instance (D, F), and show how to obtain a reduced instance (B, F') where B is a block of D, by safely colouring $V(D) \setminus V(B)$. Then, (B, F') is a hard pair if and only if (D, F) is, in which case we may terminate the algorithm. Otherwise, the following subsections show that B may be F'-dicoloured, and together with the colouring of $V(D) \setminus V(B)$ fixed at this step, this yields an F-dicolouring of D.

Our reduction of (D, F) proceeds by considering the end-blocks of D one after the other. If an end-block B, together with F, may correspond to a monochromatic hard pair, a bicycle hard pair, or a complete hard pair glued to the rest of the digraph, then we safely colour $V(B) \setminus V(D)$ and move to the next end-block. If B cannot be such a block, then we will show that we can safely colour $V(D) \setminus V(B)$.

Before formalising this strategy, we need a few definitions. Given a valid pair (D, F), an end-block B of D, with cut-vertex x, is a *hard end-block* if it is of one of the following types:

(*i*) There exists a colour $i \in [s]$ such that $f_i(x) \geq (d_B^-(x), d_B^+(x))$, and for every $v \in V(B) \setminus \{x\}$ we have $f_i(v) = (d_B^-(v), d_B^+(v))$ and $f_k(v) = (0, 0)$ when $k \neq i$.

We refer to such a hard end-block as a *monochromatic hard end-block*.

(*ii*) B is a bidirected odd cycle and there exists colours $i \neq j$ such that, $f_i(x) \geq (1, 1)$, $f_j(x) \geq (1, 1)$ and for every $v \in V(B) \setminus \{x\}$, we have $f_c(v) = (1, 1)$ if $c \in \{i, j\}$ and $f_c(v) = (0, 0)$ otherwise.

We refer to such a hard end-block as a *bicycle hard end-block*.

(*iii*) B is a bidirected complete graph, the functions $f_1, \ldots, f_s$ are constant and symmetric on $V(B) \setminus \{x\}$, and $f_i(x) \geq f_i(u)$ for every $i \in [s]$ and every $u \in V(B) \setminus \{x\}$.

We refer to such a hard end-block as a *complete hard end-block*.

Let $(D, F = (f_1, \ldots, f_s))$ be a tight valid pair such that D is not biconnected and let B be a hard end-block of D with cut-vertex x. Let u be any vertex of B distinct from x. We define the *contraction* $(D', F' = (f_1', \ldots, f_s'))$ of (D, F) with respect to B, as follows:

- $D' = D - (V(B) \setminus \{x\})$;
- for every vertex $v \in V(D') \setminus \{x\}$ and every $i \in [s]$, set $f_i'(v) = f_i(v)$;
- if B is a monochromatic hard end-block, let c be the unique colour such that $f_c(u) \neq (0, 0)$, then set $f_c'(x) = f_c(x) - (d_B^-(x), d_B^+(x))$ and $f_i'(x) = f_i(x)$ for every $i \in [s] \setminus \{c\}$;
- otherwise, B a bicycle or a complete hard end-block, and we set $f_i'(x) = f_i(x) - f_i(u)$ for every $i \in [s]$.

Lemma 5.3.6. *Let $(D, F = (f_1, \ldots, f_s))$ be a tight valid pair such that D is not biconnected and let B be a hard end-block of D with cut-vertex x. Let $(D', F' = (f_1', \ldots, f_s'))$ be the contraction of (D, F) with respect to B. Then (D, F) is a hard pair if and only if (D', F') is a hard pair.*

Proof. For the backwards direction, if (D', F') is a hard pair, then (D, F) can be defined as the join hard pair obtained from two hard pairs, (D', F') and $(B, \tilde{F} = (\tilde{f}_1, \ldots, \tilde{f}_s))$, where $\tilde{f}_i = f_i - f_i'$ for every $i \in [s]$.

Conversely, let us show by induction on the number of blocks that for every hard pair (D, F), and every end-block B, B is a hard end-block and (D', F'), the contraction of (D, F) with respect to B, is a hard pair. This is trivial if D is biconnected as D does not have any end-block. We thus assume that (D, F) is a join hard pair obtained from two hard pairs (D_1, F_1) and (D_2, F_2). Assume without loss of generality that B is a block of D_1. If this is not possible, since B is an end-block of D, we necessarily have $D_1 = B$. Hence, (D', F') is exactly (D_2, F_2), which is then biconnected, and a hard pair. Furthermore, irrespective of the type of hard pair (D_1, F_1), by the definition of hard joins, $D_1 = B$ is clearly a hard end-block of D. If B is an end-block of D_1, by the induction hypothesis, B is a hard end-block of (D_1, F_1) and the contraction of (D_1, F_1) with respect to B, (D_1', F_1'), is a hard pair. It is easy to check that (D', F') is exactly the join hard pair obtained from (D_1', F_1') and (D_2, F_2). □

Before describing our algorithm reducing the instance to a single block, we need the following subroutine testing if an end-block is hard.

Lemma 5.3.7. *Given a tight valid pair $(D, F = (f_1, \dots, f_s))$, and an end-block B of D with cut-vertex x, testing whether B is a hard end-block can be done in time $O(|V(B)| + |A(B)|)$.*

Proof. The algorithm takes block B with its cut-vertex x, and considers any $u \in V(B) \setminus \{x\}$ as a reference vertex.

We first check whether B is a monochromatic hard end-block. To do so, we first check whether exactly one colour i is available for u (*i.e.* $\{j \mid f_j(u) \neq (0,0)\} = \{i\}$). We then check whether i is indeed the only available colour for every vertex $v \in V(B) \setminus \{u, x\}$. We finally check whether $f_i(x) \geq (d^-_D(x), d^+_D(x))$. Since (D, F) is tight, B is a monochromatic hard end-block if and only if all these conditions are met.

Assume now that B is not a monochromatic hard end-block. Now, B is a hard end-block if and only if it is a bicycle hard end-block or a complete hard end-block. In both cases, the functions $f_1, \dots, f_s$ must be symmetric, and constant on $V(B) \setminus \{x\}$. Since (D, F) is tight, for every vertex $v \in V(B) \setminus \{x\}$, we can iterate over $\{j \mid f_j(v) \neq (0,0)\}$ in time $O(d(v))$. This allows us to check in linear time whether the functions $f_1, \dots, f_s$ are symmetric and constant on $V(B) \setminus \{x\}$. If this is not the case, B is not a hard end-block. We can assume now that $f_1, \dots, f_s$ are symmetric and constant on $V(B) \setminus \{x\}$. We also assume that B is either a bidirected odd cycle or a bidirected complete graph, for otherwise B is clearly not a hard end-block.

If B is a bidirected odd cycle, we check in constant time that $f_i(u) \neq (0,0)$ holds for exactly two colours. If this does not hold, the end-block B is not a hard end-block. Otherwise, denote these two colours i, j, and note (by tightness and symmetry) that $f_i(u) = f_j(u) = (1,1)$, and that $f_k(u) = (0,0)$ for $k \notin \{i, j'\}$. Now B is a bicycle hard end-block if and only if $f_k(x) \geq (1,1)$ for $k \in \{i, j\}$, which can be checked in constant time.

Assume finally that B is a bidirected complete graph. Since (D, F) is tight, and because the functions $f_1, \dots, f_s$ are symmetric and constant on $V(B) \setminus \{x\}$, then B is a complete hard end-block if and only if $f_i(x) \geq f_i(u)$ for every colour $i \in [s]$. This can be checked in $O(|V(B)|)$ time, since there are at most $|V(B)|$ colours i such that $f_i(u) \neq (0,0)$ (as (D, F) is tight). □

We now turn to showing the main lemma of this subsection, which reduces any tight valid pair to a biconnected one.

Lemma 5.3.8. *Let (D, F) be a tight valid pair. There exists a block B of D such that $V(D) \setminus V(B)$ may be safely coloured, yielding the reduced pair (B, F'). Moreover, (B, F') and the colouring of $V(D) \setminus V(B)$ can be computed in linear time.*

Proof. Let (D, F) be such a pair. We first compute, in linear time [TV85], an ordering $B^1, \dots, B^r$ of the blocks of D and an ordered set of vertices $(x_2, \dots, x_r)$ in such a way that whenever $2 \leq \ell \leq r$, the only cut-vertex of $D^\ell = D[\bigcup_{j=1}^{\ell} V(B^\ell)]$ in B^ℓ is x_ℓ. If D is biconnected, that is $r = 1$, the result follows for $B = D$ and there is nothing to do. We now assume $r \geq 2$.

For ℓ going from r to 2, we proceed as follows. We consider the block B^ℓ, which is an end-block of D^ℓ. We will either safely colour $V(D^\ell) \setminus V(B^\ell)$ and output the reduced pair (B^ℓ, F'), or safely colour $V(B^\ell) \setminus \{x_\ell\}$, compute the reduced pair $(D^{\ell-1}, F^{\ell-1})$, and go on with the end-block $B^{\ell-1}$ of $D^{\ell-1}$. If we colour all the blocks B^ℓ with $\ell \geq 2$, we output the reduced pair (D^1, F^1).

More precisely, when considering B^ℓ, we first check in time $O(|V(B^\ell)| + |A(B^\ell)|)$ whether B^ℓ is a hard end-block of (D^ℓ, F^ℓ) (by Lemma 5.3.7). We distinguish two cases, depending on whether B^ℓ is a hard end-block.

Case 1: *B^ℓ is a hard end-block of D^ℓ.*

In this case, we safely colour the vertices of $V(B^\ell) \setminus \{x_\ell\}$ in time $O(|V(B^\ell)| + |A(B^\ell)|)$. To do so, we first compute an ordering $\sigma = (v_1, \dots, v_b = x_\ell)$ of $V(B^\ell)$ corresponding to a leaves-to-root ordering of a spanning tree of B^ℓ rooted in x_ℓ, in time $O(|V(B^\ell)| + |A(B^\ell)|)$. We then greedily colour the vertices $v_1, \dots, v_{b-1}$ in this order. For every $j \in [b-1]$, the vertex v_j has at least one neighbour in $\{v_{j+1}, \dots, v_b\}$ (its parent in the spanning tree), so condition ($\star\star$) is fulfilled. Hence, Lemma 5.3.3 ensures that the algorithm succeeds and that the reduced pair $(D^{\ell-1}, F^{\ell-1})$ is a valid pair. We will now show that $(D^{\ell-1}, F^{\ell-1})$ is necessarily tight, and that the colouring of $V(B^\ell) \setminus \{x_\ell\}$ we computed is safe, by showing that $(D^{\ell-1}, F^{\ell-1})$ is exactly the contraction of (D^ℓ, F^ℓ) with respect to B^ℓ.

Assume first that B^ℓ is a monochromatic hard end-block. Then, all the vertices of $V(B) \setminus \{x_\ell\}$ are coloured with the same colour i, and by the definition of a monochromatic hard end-block, we have $f_i^{\ell-1}(x_\ell) = f_i^\ell(x_\ell) - (d^-_{B^\ell}(x_\ell), d^+_{B^\ell}(x_\ell))$ and $f_j^{\ell-1}(x_\ell) = f_j^\ell(x_\ell)$ for every colour $j \neq i$. Hence, $(D^{\ell-1}, F^{\ell-1})$ is tight (as the degrees from D^ℓ to $D^{\ell-1}$ only change for x_ℓ, for which they decrease exactly by $d^-_{B^\ell}(x_\ell)$, and $d^+_{B^\ell}(x_\ell)$) and Lemma 5.3.6 implies that the colouring we computed is safe.

Assume now that B^ℓ is a bicycle hard end-block. Let $i \neq j$ be such that $f_i^\ell(u) = f_j^\ell(u) = (1,1)$ for every vertex $u \in V(B^\ell) \setminus \{x_\ell\}$. Recall that, by the definition of a bicycle hard end-block, $f_k^\ell(x_\ell) \geq (1,1)$ for $k \in \{i,j\}$. Let u, v be the two neighbours of x_ℓ in B^ℓ, then u and v are connected by a bidirected path of odd length in $B^\ell - x_\ell$. This implies that u and v are coloured differently, and $f_k^{\ell-1}(x_\ell) = f_k^\ell(x_\ell) - (1,1)$ for $k \in \{i,j\}$ and $f_k^{\ell-1}(x_\ell) = f_k^\ell(x_\ell)$ for $k \notin \{i,j\}$. Since $d^+_{B^\ell}(x_\ell) = d^-_{B^\ell}(x_\ell) = 2$, know that $(D^{\ell-1}, F^{\ell-1})$ is tight, and Lemma 5.3.6 implies that the colouring we computed is safe.

Assume finally that B^ℓ is a complete hard end-block, and let u be any neighbour of x_ℓ in B. The functions $f_1^\ell, \dots, f_s^\ell$ are constant and symmetric on $V(B) \setminus \{x_\ell\}$. We also have $f_i^\ell(x_\ell) \geq f_i^\ell(u)$ for every $i \in [s]$. By construction of the greedy colouring, for every $i \in [s]$, at most $f_i^{\ell+}(u)$ vertices of $V(B^\ell) \setminus \{x_\ell\}$ are coloured i. Since $|V(B^\ell)| - 1$ vertices are coloured in total, and because $\sum_{i=1}^s f_i^{\ell+}(u) = |V(B^\ell)| - 1$, we conclude that, for every $i \in [s]$, exactly $f_i^{\ell+}(u)$ vertices of $V(B^\ell) \setminus \{x_\ell\}$ are coloured i. Therefore, $(D^{\ell-1}, F^{\ell-1})$ is tight, and Lemma 5.3.6 implies that the colouring we computed is safe.

Case 2: B^ℓ is not a hard end-block of D^ℓ.

In this case, we colour the vertices of $V(D^{\ell-1}) \setminus \{x_\ell\}$ in time $O(|V(D)| + |A(D)|)$ as follows. We first compute an ordering $\sigma = (v_1, \dots, v_{n'} = x_\ell)$ of $V(D^{\ell-1})$, that is a leaves-to-root ordering of a spanning tree of $D^{\ell-1}$ rooted in x_ℓ, in time $O(|V(D^\ell)| + |A(D^\ell)|)$. We greedily colour vertices $v_1, \dots, v_{n'-1}$ in this order. For every $j \in [n'-1]$, the vertex v_j has at least one neighbour in $\{v_{j+1}, \dots, v_{n'}\}$ (its parent in the spanning tree), so condition ($\star\star$) is fulfilled. Hence by Lemma 5.3.3, the greedy colouring succeeds and provides a reduced pair $(B^\ell, F' = (f'_1, \dots, f'_s))$ that is valid.

Finally observe that (B^ℓ, F') is not a hard pair since, for every $i \in [s]$, we have $f'_i(x_\ell) \leq f_i^\ell(x_\ell)$. So if (B^ℓ, F') is a hard pair, B^ℓ is necessarily a hard end-block of (D^ℓ, F^ℓ), a contradiction. The result follows.

Note that Case 2 may only be reached once, at which point it outputs a reduced pair. Therefore, the running time of the algorithm described above is bounded by $O(\sum_\ell \left(|V(B^\ell)| + |A(B^\ell)|\right) + |V(D)| + |A(D)|)$, which is linear in $|V(D)| + |A(D)|$. Assume finally that, in the process above, the second case is never attained. Then the result follows as B^1 is a block of D, and we found a safe colouring of $V(D) \setminus V(B^1)$. □

With the last two lemmas at our disposal, we are ready to test whether a tight valid pair is hard.

Lemma 5.3.9. *Given a tight valid pair $(D, F = (f_1, \dots, f_s))$, testing whether (D, F) is a hard pair can be done in time linear time.*

Proof. We first consider the pair (B, F') reduced from (D, F), where B is a block of D, obtained by safely colouring $V(D) \setminus V(B)$ through Lemma 5.3.8. In particular, (B, F') is a hard pair if and only if (D, F) is, and as B is biconnected, we may only check whether (B, F') is a biconnected hard pair, either monochromatic, bicycle or complete. This amounts to verifying the conditions of the definition. The fact that this can be done in linear time is already justified in Lemma 5.3.7. Indeed, testing (B, F') for a hard biconnected pair amounts to testing the conditions for a hard end-block, except for the condition on the cut-vertex. Since there is no cut-vertex in B, we test the same conditions on all vertices of B. □

5.3.6 Reducing to pairs with two colours

We have just shown how to reduce any tight valid pair into a biconnected one, after which we may decide whether the initial pair was a hard one. In this subsection, we therefore consider any tight valid pair (D, F) that is not hard, and such that D is biconnected. We show how the problem of F-dicolouring D boils down to $\tilde{F}$-dicolouring D where $\tilde{F}$ involves only $s = 2$ colours.

Lemma 5.3.10. *Let (D, F) be a valid pair that is not a hard pair and such that D is biconnected. In linear time, we can either find an F-dicolouring of D or compute a valid pair $(D, \tilde{F} = (\tilde{f}_1, \tilde{f}_2))$ such that:*

- *$(D, \tilde{F})$ is not hard, and*
- *given an $\tilde{F}$-dicolouring of D, we can compute an F-dicolouring of D in linear time.*

Proof. We assume that (D, F) is tight, otherwise we compute an F-dicolouring of D in linear time by Lemma 5.3.5. Since (D, F) is tight, we can iterate over $\{i \mid f_i(v) \neq (0, 0)\}$ in time $O(d(u))$. This allows us to check, in linear time, which of the cases below apply to (D, F).

Case 1: *There exist $u \in V(D)$ and $i \in [s]$ such that $f_i^+(u) \neq f_i^-(u)$.*

We define $\tilde{f}_1 = f_i$ and $\tilde{f}_2 = \sum_{j \in [s], j \neq i} f_j$. Let us show that $(D, \tilde{F})$ is not a hard pair. Since D is biconnected, $(D, \tilde{F})$ is not a join hard pair. Since $\tilde{f}_1$ is not symmetric by the choice of $\{u, i\}$, $(D, \tilde{F})$ is neither a bicycle hard pair nor a complete hard pair. Finally, assume for a contradiction that $(D, \tilde{F})$ is a monochromatic hard pair. Since $\tilde{f}_1(u) \neq (0, 0)$, we thus have $\tilde{f}_2(v) = (0, 0)$ and $f_i(v) = (d^-(v), d^+(v))$ for every vertex v. We conclude that (D, F) is also a monochromatic hard pair, a contradiction.

Case 2: *There exist $u, v \in V(D)$ and $i \in [s]$ such that $f_i(u) \neq f_i(v)$.*

As in Case 1, we set $\tilde{f}_1 = f_i$ and $\tilde{f}_2 = \sum_{j \in [s], j \neq i} f_j$. Since D is biconnected, $(D, \tilde{F} = (\tilde{f}_1, \tilde{f}_2))$ is not a join hard pair. Moreover, since $f_i(v) \neq f_i(u)$, $\tilde{f}_1$ is not constant so $(D, \tilde{F})$ is neither a bicycle hard pair nor a complete hard pair. Again, assume for a contradiction that $(D, \tilde{F})$ is a monochromatic hard pair. Since $\tilde{f}_1(u)$ or $\tilde{f}_1(v)$ is distinct from $(0,0)$, we thus have $\tilde{f}_2(v) = (0,0)$ and $f_i(v) = (d^-(v), d^+(v))$ for every vertex v. We conclude that (D,F) is also a monochromatic hard pair, a contradiction.

Case 3: *None of the cases above is matched.*

Therefore, for each $i \in [s]$, f_i is a symmetric constant function. Thus, since (D,F) is not a hard pair, and because (D,F) is tight, D is not a bidirected odd cycle or a bidirected complete graph. Let $i \in [s]$ be such that f_i is not the constant function equal to $(0,0)$. We set $\tilde{f}_1 = f_i$ and $\tilde{f}_2 = \sum_{j \in [s], j \neq i} f_j$. Since (D,F) is not a monochromatic hard pair, there is a colour $k \neq i$ such that f_k is not the constant function equal to $(0,0)$. Hence, none of $\tilde{f}_1, \tilde{f}_2$ is the constant function equal to $(0,0)$, and $(D, (\tilde{f}_1, \tilde{f}_2))$ is not a hard pair.

In each case, we have built a valid pair $(D, \tilde{F} = (\tilde{f}_1, \tilde{f}_2))$ where $\tilde{f}_1 = f_i$ and $\tilde{f}_2 = \sum_{j \neq i} f_j$, in such a way that $(D, \tilde{F})$ is not hard. We finally prove that, given an $\tilde{F}$-dicolouring of D, we can compute an F-dicolouring of D in linear time. Let $\tilde{\alpha}$ be an $\tilde{F}$-dicolouring of D. Let X be the set of vertices coloured 1 in $\tilde{\alpha}$ and $\hat{D}$ be $D - X$ ($\hat{D}$ is built in linear time by successively removing vertices in X from D). We define $\hat{F} = (\hat{f}_1, \ldots, \hat{f}_s)$, $\hat{f}_i : V(\hat{D}) \to \mathbb{N}^2$ as follows:

$$\hat{f}_k(v) = \begin{cases} f_k(v) & \text{if } k \neq i \\ (0,0) & \text{otherwise.} \end{cases}$$

Since $\tilde{f}_2$ is exactly $\sum_{i=1}^{s} \hat{f}_i$, by the definition of $\tilde{\alpha}$ and $\hat{D}$, we know that $\hat{D}$ is strictly-$(\sum_{i=1}^{s} \hat{f}_i)$-bidegenerate. Then applying Lemma 5.3.4 yields an $\hat{F}$-dicolouring $\hat{\alpha}$ of $\hat{D}$ in linear time. The colouring α defined as follows is thus an F-dicolouring of D obtained in linear time:

$$\alpha(v) = \begin{cases} i & \text{if } \tilde{\alpha}(v) = 1 \\ \hat{\alpha}(v) & \text{otherwise.} \end{cases}$$

□

5.3.7 Solving blocks with two colours - particular cases

We are now left to deal with pairs (D,F) such that D is biconnected, and F only involves two colours. Eventually, our strategy consists in finding a suitable decomposition for D, and colouring parts of it inductively, which will be done in Subsection 5.3.8. In this subsection, we first deal with some specific forms (D,F) may take, showing D can be F-dicoloured in those cases. Along the way, this allows us to deal with increasingly restricted instances, later enabling us to constrain the instances considered in the induction. In Lemma 5.3.11 and Lemma 5.3.12, we exhibit an F-dicolouring of D if there exists a vertex x which does not satisfy some conditions relating its neighbourhood and F. Then, we solve instances where D is a bidirected complete graph in Lemma 5.3.13, or an orientation of a cycle in Lemma 5.3.14. Lastly, we solve those that are constructed by a star attached to a cycle in Lemma 5.3.15. These will serve as base cases for the induction.

Given a valid pair $(D, F = (f_1, f_2))$ that is tight, non-hard, and such that D is biconnected, let us define a set of properties that (D,F) may or may not fulfil. Each of these properties is easy to check in linear time, and we will see in the following that when they are not met, there is a linear-time algorithm providing an F-dicolouring of D. The first property, (E), guarantees that both f_1 and f_2 *exceed* some lower bound. It states that both f_1^-, f_2^- (respectively f_1^+, f_2^+) are non-zero on vertices having at least one in-neighbour (respectively out-neighbour).

$$\forall x, c \quad \text{we have} \quad (d^-(x) > 0 \Longrightarrow f_c^-(x) \geq 1) \quad \text{and} \quad (d^+(x) > 0 \Longrightarrow f_c^+(x) \geq 1). \tag{E}$$

Given a digraph $D = (V, A)$, we define the function $\mathbb{1}_A : V \times V \to \mathbb{N}$ as follows:

$$\mathbb{1}_A(u, v) = \begin{cases} 1 & \text{if } (u, v) \in A \\ 0 & \text{otherwise.} \end{cases}$$

Note in particular, that if (E) holds, then the following holds.

$$\forall x, y, z, c \quad \text{we have} \quad f_c(x) \geq (\mathbb{1}_A(y, x), \mathbb{1}_A(x, z)). \tag{E'}$$

Lemma 5.3.11. *Let $(D, F = (f_1, f_2))$ be a valid tight non-hard pair such that D is biconnected. If the pair does not fulfill property* (E)*, then D is F-dicolourable. Furthermore, there is a linear-time algorithm that checks property* (E)*, and if the property is not met, computes an F-dicolouring of D.*

Proof. We first prove that, for some arc (u, v) and for some colour $c \in \{1, 2\}$, we have:

$$(f_c^+(u) = 0 \land f_c(v) \neq (0, 0)) \quad \text{or} \quad (f_c^-(v) = 0 \land f_c(u) \neq (0, 0)). \tag{1}$$

Note that, if it exists, such an arc is found in linear time. Assume for a contradiction that no such arc exists. By assumption, $\exists x \in V(D),\ c \in \{1, 2\}$ such that $(d^+(x) > 0 \land f_c^+(x) = 0)$ or $(d^-(x) > 0 \land f_c^-(x) = 0)$. If $(d^+(x) > 0 \land f_c^+(x) = 0)$, and y is any out-neighbour of x, then we must have $f_c(y) = (0, 0)$ for otherwise (x, y) clearly satisfies (1). Symmetrically, if $(d^-(x) > 0 \land f_c^-(x) = 0)$, and y is any in-neighbour of x, then we have $f_c(y) = (0, 0)$ for otherwise (y, x) satisfies (1). In both cases, we conclude on the existence of a vertex y for which $f_c(y) = (0, 0)$. Let Y be the non-empty set of vertices y for which $f_c(y) = (0, 0)$ and Z be $V(D) \setminus Y$, that is the set of vertices z for which $f_c(z) \neq (0, 0)$. Since (D, F) is valid and tight, Z must also be non-empty, for otherwise (D, F) is a monochromatic hard pair. The connectivity of D guarantees the existence of an arc between a vertex in Y and another one in Z. This arc satisfies (1).

Once we found an arc (u, v) satisfying (1), we proceed as follows to compute an F-dicolouring of D. If $f_c^+(u) = 0 \land f_c(v) \neq (0, 0)$, we colour v with c, otherwise we have $f_c^-(v) = 0 \land f_c(u) \neq (0, 0)$ and we colour u with c. Let $(D', F' = (f'_1, f'_2))$ be the pair reduced from this colouring and let c' be the colour distinct from c. In the former case, that is v is coloured with c, we obtain ${f'_{c'}}^+(u) = f_{c'}^+(u) \geq d_D^+(u) > d_{D'}^+(u)$. In the latter case, u is coloured with c and ${f'_{c'}}^-(v) = f_{c'}^-(v) \geq d_D^-(v) > d_{D'}^-(v)$. In both cases, (D', F') is loose, meaning the result follows from Lemma 5.3.5. ☐

The next property, (DS), guarantees that the vertices incident only to digons have symmetric constraints.

$$\forall x, c \quad \text{we have} \quad N^-(x) \neq N^+(x), \quad \text{or} \quad f_c^-(x) = f_c^+(x). \tag{DS}$$

Again, we may solve the instance at this point if (D, F) does not satisfy (DS).

Lemma 5.3.12. *Let $(D, F = (f_1, f_2))$ be a valid tight non-hard pair such that D is biconnected. If the pair does not fulfill property* (DS)*, then D is F-dicolourable. Furthermore, there is a linear-time algorithm checking property* (DS)*, and if the property is not met, computing an F-dicolouring of D.*

Proof. Recall that the data structure encoding D allows checking $N^+(u) = N^-(u)$ in constant time (as this is equivalent to checking $|N^+(u) \cap N^-(u)| = d^-(u) = d^+(u)$), so in linear time we may find a vertex x such that $N^-(x) = N^+(x)$ and $f_c^-(x) \neq f_c^+(x)$ for some $c \in \{1, 2\}$.

Let T be a spanning tree rooted in x and let $\sigma = (v_1, \ldots, v_n = x)$ be a leaves-to-root ordering of $V(D)$ with respect to T. For every $j \in [n-1]$, the vertex v_j has at least one neighbour in $\{v_{j+1}, \ldots, v_n\}$ (its parent in the spanning tree), so condition $(\star\star)$ is fulfilled. Hence by Lemma 5.3.3, there is an algorithm that computes an F-dicolouring of $D - x$. It remains to show that the reduced pair (D', F') is non-hard. As D' is the single vertex x, this is equivalent to showing that $f'_c(x) \neq (0, 0)$ for some c.

Towards a contradiction, suppose $f'_1(x) = f'_2(x) = (0, 0)$. For $i \in \{1, 2\}$, let d_i be the number of neighbours of x coloured i, and note that $d_1 + d_2 = |N(x)|$. Since $f_i(x) - (d_i, d_i) \leq f'_i(x) = (0, 0)$ we have:

$$\forall i \qquad f_i(x) \leq (d_i, d_i). \tag{2}$$

By tightness, we then have:

$$(|N(x)|, |N(x)|) = f_1(x) + f_2(x) \leq (d_1 + d_2, d_1 + d_2) = (|N(x)|, |N(x)|)$$

Thus, the inequalities of (2) are equalities, and both f_1, f_2 are symmetric on x, a contradiction. □

At this point, we have proven that if one of (E) or (DS) does not hold, an F-dicolouring of D may be computed in linear time. From now on, we thus consider only instances satisfying both conditions, and move on to solving the base cases of our induction, corresponding to D having a certain structure. We first prove that if D is a bidirected complete graph, and the instance is not hard, an F-dicolouring of D exists and can be computed in linear time.

Lemma 5.3.13. *Let $(D, F = (f_1, f_2))$ be a valid tight non-hard pair such that D is a bidirected complete graph. Then D is F-dicolourable and there is a linear-time algorithm providing an F-dicolouring of D.*

Proof. By Lemma 5.3.12, we can assume that f_1, f_2 are symmetric (i.e. $\forall x, c\ f_c^+(x) = f_c^-(x)$). Let v be any vertex such that $f_1^+(v) = \min\{f_1^+(x) \mid x \in V(D)\}$. Let u be any vertex such that $f_1^+(u) > f_1^+(v)$. The existence of u is guaranteed, for otherwise the tightness of (D, F) implies that (D, F) is a complete hard pair, a contradiction. Let $X \subseteq (V(D) \setminus \{u, v\})$ be any set of $f_1^+(v)$ vertices (we have $f_1^+(u) \leq n - 1$, which implies $f_1^+(v) \leq n - 2$, and the existence of X is guaranteed). Note that X, u, and v can be computed in linear time. We colour all the vertices of X with 1, and note that this is an F-dicolouring of $D[X]$, as every vertex $x \in X$ satisfies $f_1^+(x) \geq f_1^+(v) = |X| > |X| - 1 = d^+_{D[X]}(x)$. Let $(D', F' = (f'_1, f'_2))$ be the pair reduced from this colouring. Observe that we may have $X = \emptyset$, in which case $(D', F') = (D, F)$. Then $f'_1(v) = (0, 0)$ and $f'_1(u) \neq (0, 0)$, so (D', F') is not a hard pair. If (D', F') is loose, we conclude with Lemma 5.3.5, otherwise may conclude with Lemma 5.3.11 since (D', F') does not fulfill (E) as witnessed by v. □

The following shows how to compute, in linear time, an F-dicolouring of D if the underlying graph of D is a cycle.

Lemma 5.3.14. *Let $(D, F = (f_1, f_2))$ be a valid tight non-hard pair such that* $\mathrm{UG}(D)$ *is a cycle. Then D is F-dicolourable and there is a linear-time algorithm providing an F-dicolouring of D.*

Proof. By Lemma 5.3.11, we can assume that (E) holds. Let us show that for any vertex $x \in V(D)$, we have $d^-(x), d^+(x) \in \{0, 2\}$. Towards a contradiction, and by symmetry, assume that some vertex x verifies $d^-(x) = 1$. By tightness and by (E), we have $1 = d^-(x) = f_1^-(x) + f_2^-(x) \geq 1 + 1$, a contradiction. We distinguish two cases, depending on whether D contains a digon or not.

Case 1: *D contains a digon.*

In this case, there is only one orientation avoiding in- or out-degree one, the one with D fully bidirected. We now claim that, by (E'), for every vertex $x \in V(D)$ and every colour c, $f_c(x) \geq (1, 1)$ Hence, by tightness f_1, f_2 are constant functions equal to $(1, 1)$. Since (D, F) is not a hard pair, we conclude that $|V(D)|$ is even, and any proper 2-colouring of $\mathrm{UG}(D)$ is indeed an F-dicolouring.

Case 2: *D does not contain any digon.*

In this case, the only possible orientation avoiding in- and out-degree one is the antidirected cycle, that is an orientation of a cycle in which every vertex is either a source or a sink. In that case we colour every vertex with colour one, and note that for a sink x (resp. a source) we have $f_1^+(x) = 1 > 0 = d^+(x)$ (resp. $f_1^-(x) = 1 > 0 = d^-(x)$). Hence, this colouring is indeed an F-dicolouring of D.

□

The following shows how to compute, in linear time, an F-dicolouring of D if the underlying graph of D is a *subwheel*, that is, a cycle with an additional vertex.

Lemma 5.3.15. *Let $(D, F = (f_1, f_2))$ be a valid tight non-hard pair, and let $v \in V$ be such that* $\mathrm{UG}(D - v)$ *is a cycle C. Then D is F-dicolourable and there is a linear-time algorithm providing an F-dicolouring of D.*

Proof. By applying Lemma 5.3.11 and Lemma 5.3.12, we may assume that (E) and (DS) hold. Let $v_1, \dots, v_{n-1}$ be an ordering of $V(D) \setminus \{v\}$ along C, which can be obtained in linear time. We distinguish several cases, according to the structure of (D, F). It is straightforward to check, in linear time, which of the following cases matches the structure of (D, F).

Case 1: $|N(v)| < |V(D)| - 1$.

Let w be any neighbour of v and let y be any vertex that is not adjacent to v. Let ρ_w be $(f_1^-(w) - \mathbb{1}_A(v, w), f_1^+(w) - \mathbb{1}_A(w, v))$. Informally, ρ_w corresponds to the new value of $f_1(w)$ if we were to colour v with 1. If $\rho_w \neq (1, 1)$, we colour v with 1, otherwise we colour v with 2. Let $(D', F' = (f'_1, f'_2))$ be the pair reduced from colouring v, we claim that (D', F') is not a hard pair.

For every $c \in \{1, 2\}$, we have $f'_c(y) = f_c(y)$ because y is not adjacent to v, and by (E) $f'_c(y) = f_c(y) \neq (0, 0)$. Therefore (D', F') cannot be a monochromatic hard pair. It is also not a join hard pair because D' is biconnected.

Assume that (D', F') is a bicycle hard pair. Hence, $f'_1(w) = (1, 1)$, so v is coloured 2, as otherwise we would have $f'_1(w) = \rho_w \neq (1, 1)$. But if v is coloured 2, then $(1, 1) = f'_1(w) =$

$f_1(w)$ and in that case $\rho_w \neq (1,1)$ (as w is adjacent to v), and we should have coloured v with colour 1, a contradiction.

Assume that (D', F') is a complete hard pair, then D' is a bidirected complete graph. Since $\mathrm{UG}(D')$ is a cycle, D' is necessarily $\overleftrightarrow{K}_3$. Such a complete hard pair is also, either a monochromatic hard pair or a bicycle hard pair, but we already discarded both cases. Hence (D', F') is not a hard pair, and the result follows from Lemma 5.3.14, as $\mathrm{UG}(D')$ is a cycle.

Case 2: *D is bidirected, $d^+(v) = |V(D)| - 1$, $|V(D)|$ is even, and f_1 is constant on $V(D) \setminus \{v\}$.*

Assume $D \neq \overleftrightarrow{K}_4$ as otherwise the result follows from Lemma 5.3.13. Observe that $D - v$ is a bidirected odd cycle, and that it contains at least 5 vertices for otherwise D is exactly $\overleftrightarrow{K}_4$. Since D is bidirected, by (DS) and (E) we have $f_c^+(x) = f_c^-(x) > 0$ for every $x \in V$ and every $c \in \{1,2\}$.

Since f_1 is constant on $V(D) \setminus \{v\}$ by assumption, so is f_2, as (D, F) is tight and as $d^-(u) = d^+(u) = 3$ for every $u \in V(D) \setminus \{v\}$. Assume without loss of generality that $f_1(u) = (1,1)$ and $f_2(u) = (2,2)$ for every $u \in V(D) \setminus \{v\}$.

If $f_1^+(v) \geq 2$, we colour v and v_1 with 1, and all other vertices with 2, see Figure 5.3(a). The digraph induced by the vertices coloured 2 is a bidirected path, which is strictly-f_2-bidegenerate since on these vertices, f_2 is the constant function equal to $(2,2)$. The digraph induced by the vertices coloured 1 is $\overleftrightarrow{K}_2$ and contains v. Since $f_1^+(v) \geq 2$ and $f_1(v_1) = (1,1)$, it is strictly-f_1-bidegenerate.

Else, we have $f_1^+(v) \leq 1$, and we colour $v \cup \{v_{2i} \mid i \in \left[\frac{|V(D)|}{2}\right]\}$ with 2 and $\{v_{2i-1} \mid i \in \left[\frac{|V(D)|-2}{2}\right]\}$ with 1, see Figure 5.3(b). Vertices coloured 1 form an independent set, so the digraph induced by them is strictly-f_1-bidegenerate (as f_1 is constant equal to $(1,1)$ on these vertices). Since $f_1^+(v) \leq 1$, and because (D, F) is valid, we have $f_2^+(v) \geq n - 2$. Since $|V(D)| \geq 6$ it implies $f_2^+(v) \geq \frac{|V(D)|}{2} + 1$. Let H be the digraph induced by the vertices coloured 2. Then the out-degree of v in H is exactly $\frac{|V(D)|}{2}$. Since $f_2^+(v) \geq \frac{|V(D)|}{2} + 1$, the digraph H is strictly-f_2-bidegenerate if and only if $H - v$ is strictly-f_2-bidegenerate. Observe that $H - v$ is made of a copy of $\overleftrightarrow{K}_2$ and isolated vertices, so it is strictly-f_2-bidegenerate since f_2 is constant equal to $(2,2)$ on $V(H) \setminus \{v\}$.

Case 3: *$N^+(v) = N^-(v) = V(D) \setminus \{v\}$ and $D - v$ is a directed cycle.*

Note that in this case, for every vertex $u \in V(D) \setminus \{v\}$ we have $d^-(u) = d^+(u) = 2$, and by (E) and by the tightness of (D, F), we have $f_1(u) = f_2(u) = (1,1)$. Since $d^+(v) \geq 3$, there is a colour $c \in \{1,2\}$ such that $f_c^+(v) \geq 2$. Assume $f_2^+(v) \geq 2$. We are now ready to give an F-dicolouring explicitly. We colour v and v_1 with 2, and all other vertices with 1, see Figure 5.4 for an illustration.

The digraph induced by the vertices coloured 1 is a directed path, which is strictly-f_1-bidegenerate since on these vertices, f_1 is the constant function equal to $(1,1)$. The digraph induced by the vertices coloured 2 is $\overleftrightarrow{K}_2$ and contains v. Since $f_2^+(v) \geq 2$ and $f_2(v_1) = (1,1)$, it is strictly-f_2-bidegenerate.

Case 4: *None of the previous cases apply.*

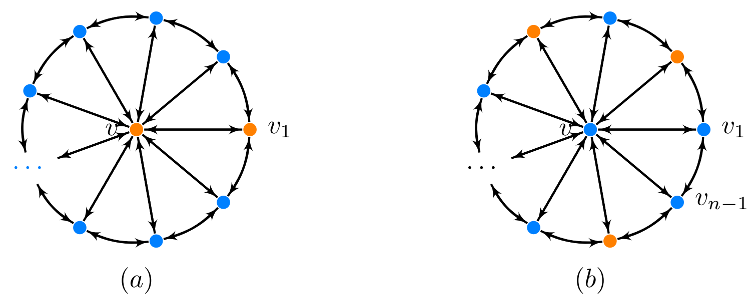


Figure 5.3: The F-dicolourings for Case 2. The partition on the left corresponds to the case $f_1^+(v) \geq 2$ and the one on the right corresponds to the case $f_1^+(v) \leq 1$. Vertices coloured 1 are represented in orange and vertices coloured 2 are represented in blue.

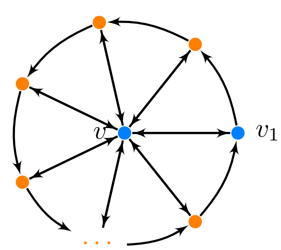


Figure 5.4: The F-dicolouring for Case 3. Vertices coloured 1 are represented in orange and vertices coloured 2 are represented in blue.

We first prove the existence of a vertex $u \neq v$ and a colour $c \in \{1,2\}$ such that $f_c(u) > (\mathbb{1}_A(v,u), \mathbb{1}_A(u,v))$. By (E'), for any vertex $u \neq v$ we have $f_c(u) \geq (\mathbb{1}_A(v,u), \mathbb{1}_A(u,v))$, we thus look for a vertex $u \neq v$ such that $f_c(u) \neq (\mathbb{1}_A(v,u), \mathbb{1}_A(u,v))$. Assume for a contradiction that such a pair does not exist, so for every vertex $x \in V(D) \setminus \{v\}$ and every colour $c \in \{1,2\}$, $f_c(x)$ is exactly $(\mathbb{1}_A(v,x), \mathbb{1}_A(x,v))$. Assume first that there exists a simple arc (x,v), then $f_1(x) = f_2(x) = (0,1)$. By (E), we deduce that x is a source. Since $\mathrm{UG}(D-v)$ is a cycle, we then have $d^+(x) = 3 > f_1^+(x) + f_2^+(x)$, a contradiction to (D,F) being a valid pair. The existence of a simple arc (v,x) is ruled out symmetrically, so we now assume $N^+(v) = N^-(v)$. Then every vertex $x \in V(D) \setminus \{v\}$ satisfies $f_1(x) = f_2(x) = (\mathbb{1}_A(v,x), \mathbb{1}_A(x,v)) = (1,1)$. Since (D,F) is a tight valid pair, this implies that $d^+(x) = d^-(x) = 2$. Hence $D - v$ is a directed cycle, so Case 3 is matched, a contradiction. This proves the existence of u and c such that $f_c(u) \neq (\mathbb{1}_A(v,u), \mathbb{1}_A(u,v))$. From now on, we assume $c = 1$ without loss of generality, and thus $f_1(u) > (\mathbb{1}_A(v,u), \mathbb{1}_A(u,v))$.

Consider the following property, which can be checked straightforwardly in linear time:

$$\exists x \in V(D) \setminus \{v\}, \quad \text{such that} \quad f_1(x) \neq (1 + \mathbb{1}_A(v,x), 1 + \mathbb{1}_A(x,v)) \quad \text{or} \quad f_2(x) \neq (1,1). \quad (3)$$

If $|V(D)|$ is odd or if (3) holds, we colour v with 1. Otherwise, we colour v with 2. In both cases, we claim that the reduced pair $(D', F' = (f'_1, f'_2))$ is not hard.

Assume first that $|V(D)|$ is odd or (3) is satisfied, so we have coloured v with 1. Hence, $f'_1(u) = f_1(u) - (\mathbb{1}_A(v,u), \mathbb{1}_A(u,v)) \neq (0,0)$ and $f'_2(u) = f_2(u) \neq (0,0)$ (by (E)), so (D', F') is not a monochromatic hard pair. If $|V(D)|$ is odd, then $|V(D')|$ is even so D' is not a bidirected odd cycle. If a vertex x satisfies (3), we have $f'_1(x) = f_1(x) - (\mathbb{1}_A(v,x), \mathbb{1}_A(x,v)) \neq (1,1)$ or $f'_2(x) = f_2(x) \neq (1,1)$. In both cases (D', F') is not a bicycle hard pair, as desired. Since $UG(D')$ is a cycle, if (D', F') were a complete hard pair it would be either a monochromatic or a bicycle hard pair, hence (D', F') is not a hard pair.

Assume finally that $|V(D)|$ is even and (3) is not satisfied. Hence v is coloured with 2 and, for every vertex $x \in V(D) \setminus \{v\}$, $f_1(x) = (1 + \mathbb{1}_A(v,x), 1 + \mathbb{1}_A(x,v))$ and $f_2(x) = (1,1)$. Since every vertex $x \neq v$ is adjacent to v, otherwise Case 1 would match, we have $f'_2(x) \neq f_2(x) = (1,1)$ so (D', F') is not a bicycle hard pair. Assume for a contradiction that it is a monochromatic hard pair. Since $f'_1(x) = f_1(x) \neq (0,0)$ for every vertex $x \neq v$, we have $f'_2(x) = (0,0)$. Since $f_2(x) = (1,1)$, we deduce that $N^+(v) = N^-(v) = V(D) \setminus \{v\}$. Hence every vertex x satisfies $f_1(x) = (2,2)$ and $f_2(x) = (1,1)$. Since (D,F) is tight, we conclude that D is bidirected, $|V(D)|$ is even and f_1 is constant on $V(D) \setminus \{v\}$. Thus Case 2 matches, a contradiction.

Since (D', F') is not a hard pair, and because $\mathrm{UG}(D')$ is a cycle, the result follows from Lemma 5.3.14.

□

5.3.8 Solving blocks with two colours - general case

The goal of this subsection is to provide an algorithm which, given a pair $(D, F = (f_1, f_2))$ that is not hard and such that D is biconnected, computes an F-dicolouring of D in linear time. The idea is to find a decomposition of the underlying graph $\mathrm{UG}(D)$, similar to an ear-decomposition, and successively colour the "ears" in such a way that the successive reduced pairs remain non-hard. Let us begin with the definition of the decomposition.

Definition 5.3.16. A *CSP-decomposition* (CSP stands for Cycle, Stars, and Paths) of a biconnected graph G is a sequence $(H_0, \ldots, H_r)$ of subgraphs of G partitioning the edges of G, such that H_0 is a cycle, and such that for any $i \in [r]$ the subgraph H_i is either:

- a star with a central vertex v_i of degree at least two in H_i, and such that $V(H_i) \cap \left(\bigcup_{0 \leq j < i} V(H_j)\right)$ is the set of leaves of H_i, or
- a path $(v_0, \ldots, v_\ell)$ of length $\ell \geq 3$, and such that $V(H_i) \cap \left(\bigcup_{0 \leq j < i} V(H_j)\right) = \{v_0, v_\ell\}$.

Lemma 5.3.17. *Every biconnected graph admits a CSP-decomposition, which can furthermore be computed in linear time.*

Proof. It is well-known that every biconnected graph G admits an ear-decomposition, and that it can be computed in linear time (see [Sch13] and the references therein). An *ear-decomposition* is similar to a CSP-decomposition except that paths of length one or two are allowed, and that there are no stars. To obtain a CSP-decomposition of G, we first compute an ear-decomposition $(H_0, \ldots, H_r)$, which we modify in order to get a CSP-decomposition. We do this in two steps.

Before describing these steps, we have to explain how a decomposition $(H_0, \ldots, H_r)$ is encoded. The sequence is encoded as a doubly-linked chain whose cells contain 1) a copy of the subgraph H_i 2) a mapping from the vertices of this copy to their corresponding ones in G, and 3) an integer n_i equal to $|\bigcup_{0 \leq j < i} V(H_j)|$. This last integer will allow us, given two cells, corresponding to H_i and H_j, to decide whether $i \leq j$ or not. Indeed, as the decomposition will evolve along the execution, maintaining indices from $\{0, \ldots, r\}$ seems hard in linear time.

The first step consists in modifying the ear-decomposition in order to get an ear-decomposition avoiding triples $i \leq j < k$ such that $H_k = (u_0, u_1)$ is of length one, where u_0, u_1 are inner vertices of H_i, H_j respectively, and where H_j is an ear of length at least three. Towards this, we first go along every ear, in increasing order, to compute $\texttt{birth}(v)$, the copy of vertex v appearing first in the current ear-decomposition. Note that $\texttt{birth}(v)$ is always an inner vertex of its ear, which may be the initial cycle. Then, we go along every ear in decreasing order, and when we have a length one ear $H_k = (u_0, u_1)$, we consider the ears of $\texttt{birth}(u_0)$ and $\texttt{birth}(u_1)$, say H_i and H_j respectively.

If $0 \leq i < j$, and H_j is a path of length at least three, we combine H_j and H_k into two ears, H' and H'', each of length at least two (see Figure 5.5 (left)). In the ear-decomposition, we insert H' and H'' at the position of H_j, and delete H_j and H_k.

If $0 < i = j$, then $H_j = H_i$ must be a path of length at least four. In that case, we combine H_j and H_k into two ears, H' and H'', each of length at least two (see Figure 5.5 (middle)). In the ear-decomposition, we insert H' and H'' at the position of H_j, and delete H_j and H_k.

If $0 = i = j$, then $H_j = H_0$ must be a cycle of length at least four. In that case, we combine H_j and H_k into two ears, H' and H'', each of length at least two (see Figure 5.5 (right)). In the ear-decomposition, we insert H' and H'' at the position of H_j, and delete H_j and H_k.

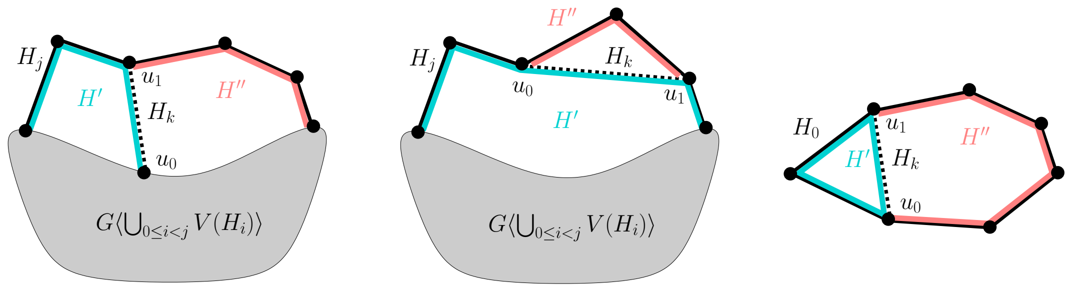


Figure 5.5: Ears H_j and H_k recombined into H' (light blue) and H'' (pink), each of length at least two. On the left, the case where H_k has one endpoint in H_j. In the middle, the case where H_k has its two endpoints in H_j. On the right, the case where H_k has its two endpoints in the cycle $H_j = H_0$.

Along this process, the ears that are modified are always shortened while still having length at least two. Hence, we do not create new couples H_j, H_k, with the second of length one, that could be recombined into ears of length at least two. This concludes the first step.

Before proceeding to the second step, recall that we now have an ear-decomposition without triples $i \le j < k$ such that $H_k = (u_0, u_1)$ is of length one, where u_0, u_1 are inner vertices of H_i, H_j respectively, and where H_j is an ear of length at least three. In particular, this implies that for any ear of length one $H_k = (u_0, u_1)$, the ears containing u_0 and u_1 as inner vertices are distinct. We denote them H_i and H_j, respectively, and assume without loss of generality that $i < j$. Note that since H_j and H_k cannot be recombined, we know H_j has length exactly two and that u_1 is its only inner vertex. The second step then simply consists, for each such ear H_k, in including H_k into H_j, redefining the latter to form a star centred in u_1.

After this process, no ear can be a path of length one. Assimilating the paths of length two as stars, we thus have a CSP-decomposition. □

We move to showing that (D, F) may be inductively coloured following this decomposition (on the underlying graph of D). Again, we only consider instances that are not dealt with in the previous subsection, which in particular satisfy (E) and (DS). Then, in the induction, Lemma 5.3.18 corresponds to colouring a star, and Lemma 5.3.19 corresponds to colouring a path, when those attach to the rest of the graph as in the decomposition. Eventually, bidirected complete graphs, cycles, and subwheels, which were solved in the previous subsection, form the base of our decomposition. The induction is formalised in Lemma 5.3.20, which achieves to solve the case of biconnected graphs with two colours.

Lemma 5.3.18. *Let $(D = (V, A), F = (f_1, f_2))$ be a valid tight non-hard pair, and let $v \in V(D)$ be such that both D and $D - v$ are biconnected, and* $\mathrm{UG}(D)$ *is neither a cycle nor a subwheel centred in v. Then there exists a colour $c \in \{1, 2\}$ such that colouring v with c is safe. Moreover, there is an algorithm that either finds c in time $O(d(v))$ or computes an F-dicolouring of D in linear time.*

Proof. By Lemma 5.3.11 and Lemma 5.3.12, we may assume that (E) and (DS) hold. Observe that $d(v) > 0$ and (E) implies that $f_1(v) \ne (0, 0)$ and $f_2(v) \ne (0, 0)$, so both colours 1 and 2 are available for v. Note also that since $\mathrm{UG}(D-v)$ is not a cycle, in particular colouring v with any colour $c \in \{1,2\}$ may never reduce it to a bicycle hard pair. We distinguish three cases.

Case 1: $|N(v)| < |V(D)| - 1$.

Let u be any neighbour of v and w be any vertex that is not adjacent to v. Since $|N(w)| > 0$, we have $f_1(w) \neq (0,0)$ and $f_2(w) \neq (0,0)$ by (E). Hence colouring v with any colour $c \in \{1,2\}$ does not reduce to a monochromatic hard pair. If $f_1(u)$ is distinct from $(f_1^-(w) + \mathbb{1}_A(v,u)), f_1^+(w) + \mathbb{1}_A(u,v)$, we colour v with 1, otherwise we colour v with 2. By the choice of u and w, the reduced pair $(D', F' = (f'_1, f'_2))$ satisfies $f'_1(u) \neq f'_1(w)$, so in particular it is not a complete hard pair.

Case 2: $|N(v)| = |V(D)| - 1$ *and* $D - v$ *is distinct from* $\overleftrightarrow{K}_{|V(D)|-1}$.

We only have to guarantee the existence of $c \in \{1,2\}$ such that colouring v with c does not reduce to a monochromatic hard pair. Also note that $|N(v)| = |V(D)|-1$ allows us $O(|V(D)|)$ time for computing c.

If there exists a vertex $x \neq v$ and a colour $c \in \{1,2\}$ such that $f_c(x) > (\mathbb{1}_A(v,x), \mathbb{1}_A(x,v))$, then colouring v with c is safe as it does not reduce to a monochromatic hard pair as $f'_c(x) = f_c(x) - (\mathbb{1}_A(v,x), \mathbb{1}_A(x,v)) \neq (0,0)$ and as $f'_{c'}(x) = f_{c'}(x) \neq (0,0)$ by (E) for the colour $c' \neq c$. Note that, if it exists, such a vertex x can be found in time $O(|V(D)|)$.

We now prove the existence of such a vertex x. Assume for a contradiction that $f_c(x) \not> (\mathbb{1}_A(v,x), \mathbb{1}_A(x,v))$ for every $x \neq v$ and every $c \in \{1,2\}$. Observe that, for every $x \neq v$, $f_c(x) \geq (\mathbb{1}_A(v,x), \mathbb{1}_A(x,v))$ by (E'), so we have $f_1(x) = f_2(x) = (\mathbb{1}_A(v,x)), \mathbb{1}_A(x,v)$. Let $x \neq v$ be any vertex. If (x,v) is a simple arc, then $f_1(x) = f_2(x) = (0,1)$ so $d^+(x) = 2$ and $d^-(x) = 0$. In particular, x has degree 1 in $\mathrm{UG}(D-v)$, a contradiction since $\mathrm{UG}(D-v)$ is biconnected and distinct from K_2 (as $\mathrm{UG}(D)$ is not a cycle). The case of (v,x) being a simple arc is symmetric, so we now assume $N^+(v) = N^-(v) = V \setminus \{v\}$, which implies $f_1(x) = f_2(x) = (1,1)$ for every vertex $x \neq v$. Hence in $D-v$ every vertex has in- and out-degree one. Since $\mathrm{UG}(D-v)$ is biconnected and distinct from K_2, $D-v$ is necessarily a directed cycle, a contradiction.

Case 3: $|N(v)| = |V(D)| - 1$ *and* $D - v$ *is* $\overleftrightarrow{K}_{|V(D)|-1}$.

If $N^+(v) = N^-(v)$ then D is $\overleftrightarrow{K}_{|V(D)|}$, and since (D,F) is not hard, Lemma 5.3.13 yields that it is F-dicolourable and an F-dicolouring is computed in linear time.

Henceforth we assume that there exists a simple arc between v and some $u \in V(D)\setminus\{v\}$. Since $D-v$ is biconnected by assumption, u must be incident to some digon, then (E) implies that $f_c(u) \geq (1,1)$ for every colour c. This guarantees that colouring v with any colour $c \in \{1,2\}$ does not reduce to a monochromatic hard pair, since in the reduced pair the constraints for u are unchanged on one coordinate. We thus have to guarantee that for some $c \in \{1,2\}$, colouring v with c does not reduce to a complete hard pair.

We assume that the simple arc between v and u goes from u to v, the other case being symmetric. If v has at least one out-neighbour w, then either $f_1^-(u) = f_1^-(w)$ and colouring v with 1 does not reduce to a complete hard pair, or $f_1^-(u) \neq f_1^-(w)$ and colouring v with 2 does not reduce to a complete hard pair.

Finally, if v is a sink and colouring v with 1 reduces to a complete hard pair, then $f_1^+(x) = f_1^-(x)+1$ for every vertex $x \neq v$. Therefore, colouring v with 2 does not reduce to a complete hard pair, as in the reduced pair we have $f_1'(x) = f_1(x)$ and $f_1^+(x) \neq f_1^-(x)$.

□

Lemma 5.3.19. *Let $(D = (V, A), F = (f_1, f_2))$ be a valid pair satisfying* (E) *and let $v_0, \dots, v_\ell$ be a path of $G = \mathrm{UG}(D)$ of length $\ell \geq 3$ such that $d_G(v_i) = 2$ for $i \in [\ell - 1]$, and such that both $G = \mathrm{UG}(D)$ and $G' = \mathrm{UG}(D - \{v_1, \dots, v_{\ell-1}\})$ are biconnected and contain a cycle. There is an algorithm computing an F-dicolouring of $D[\{v_1, \dots, v_{\ell-1}\}]$ in time $O(\ell)$ in such a way that the reduced pair (D', F') is not hard.*

Proof. Since (E) is satisfied, in particular, for any vertex x and colour c, we have $f_c(x) \neq (0, 0)$. Note that $V \setminus \{v_0, \dots, v_\ell\} \neq \emptyset$ for otherwise G' does not contain a cycle, a contradiction. Let u be any vertex in $V \setminus \{v_0, \dots, v_\ell\}$. If $f_1(v_0) = f_1(u)$, we colour v_1 with 1, otherwise we colour v_1 with 2. We then (greedily) colour $v_2, \dots, v_{\ell-1}$ using the algorithm of Lemma 5.3.3. For every $j \in \{2, \dots, \ell-1\}$, the vertex v_j has at least one neighbour in $\{v_{j+1}, \dots, v_\ell\}$, namely v_{j+1}, so condition ($\star\star$) is fulfilled, and we are ensured that the obtained colouring is an F-dicolouring $D[\{v_1, \dots, v_{\ell-1}\}]$.

Let $(D', F' = (f_1', f_2'))$ be the pair reduced from this colouring. As D' is biconnected, (D', F') is not a join hard pair. By construction, we have $f_1'(v_0) \neq f_1'(u)$ as v_0 is adjacent to v_1 but u is not adjacent to any v_i with $i \in [1, \ell - 1]$, which we just coloured. Hence (D', F') is neither a bicycle hard pair nor a complete hard pair, as f_1' is not constant on $V(D')$. Finally, (E) ensures that $f_1'(u) = f_1(u) \neq (0, 0)$ and $f_2'(u) = f_2(u) \neq (0, 0)$. Hence (D', F') is not a monochromatic hard pair. □

With these lemmas in hand we are ready to prove the main result of this subsection, that non-hard biconnected pairs are F-dicolourable.

Lemma 5.3.20. *Let $(D = (V, A), F = (f_1, f_2))$ be a valid non-hard pair. There exists an algorithm providing an F-dicolouring of D in linear time.*

Proof. We denote by $\mathcal{P}$ the property of (D, F) being tight, fulfilling (E) and (DS), not being a bidirected complete graph.

We first record the number of vertices and arcs of D, which we will keep updated for reduced pairs defined throughout the algorithm. Then, we check in linear time that each property defining $\mathcal{P}$ holds, and if one does not, we may conclude thanks to one of Lemmas 5.3.5, 5.3.11, 5.3.12 and 5.3.13.

Then, we compute a CSP-decomposition $(H_0, \dots, H_r)$ of $\mathrm{UG}(D)$ in linear time, which is possible by Lemma 5.3.17. If $r = 0$ then $\mathrm{UG}(D)$ is a cycle and the result follows from Lemma 5.3.14, assume now that $r \geq 1$.

For each i going from r to 2, we proceed as follows (if $r = 1$ we skip this part). First, if H_i is a path of length $\ell \geq 3$, we colour $v_1, \dots, v_{\ell-1}$ in time $O(\ell)$ in such a way that the reduced pair (D', F') is not hard, which is possible by Lemma 5.3.19. We also update $|V(D')|$ and $|A(D')|$, which takes time $O(\ell)$. Since $\ell \geq 3$, we can check in constant time that (D', F') satisfies $\mathcal{P}$. Indeed, properties (E) and (DS) need only be checked for v_0 and v_ℓ, which takes constant time. Then, we may check whether D' is a biclique by simply checking whether $|A(D')| = |V(D')|^2$. Then, if (D', F') does not satisfy $\mathcal{P}$, we conclude with one of Lemmas 5.3.5, 5.3.11, 5.3.12 and 5.3.13. Second, if H_i is a star with central vertex v, then we compute a safe colouring of v in time $O(d(v))$, which is possible by Lemma 5.3.18. Note that, at this step, we may directly find an F-dicolouring in linear time, in which case we stop here. Assume we do not stop, then we obtain a reduced pair (D', F'), again updating the number of arcs and vertices in time $O(d(v))$. We then check in time $O(d(v))$ that (D', F') satisfies $\mathcal{P}$ (we only have to check that it still holds for the neighbours of v in H_i). Again, if it does not, we conclude with one of Lemmas 5.3.5, 5.3.11, 5.3.12 and 5.3.13.

Assume we have not already found an F-dicolouring by the end of this process, and consider H_1. Now, Lemma 5.3.15 lets us assume that H_1 is not a star. Then, H_1 is a path, and we colour it as in the process above, then we conclude by colouring H_0 through Lemma 5.3.14.

Since $(H_0, \dots, H_r)$ partitions the edges of $\mathrm{UG}(D)$, the total running time of the described algorithm is linear in the size of D. □

5.3.9 Proof of Theorem 5.1.3

We are now ready to prove Theorem 5.1.3, that we first recall here for convenience.

Theorem 5.1.3. *Let (D, F) be a valid pair. Then D is F-dicolourable if and only if (D, F) is not a hard pair (as defined in Subsection 5.2.1). Moreover, there is an algorithm running in time $O(|V(D)| + |A(D)|)$ that decides if (D, F) is a hard pair, and that outputs an F-dicolouring if it is not.*

Proof. Let (D, F) be a valid pair. We first check in linear time whether (D, F) is tight. If it is not, we may conclude with Lemma 5.3.5. Henceforth assume that (D, F) is tight. We then check whether (D, F) is a hard pair, which is possible in linear time by Lemma 5.3.9. If (D, F) is a hard pair, then D is not F-dicolourable by Lemma 5.2.1 and we can stop. Then, we may assume that (D, F) is not a hard pair. In linear time, we find a block B of D and compute a safe colouring α of $V(D) \setminus V(B)$, which is possible by Lemma 5.3.8. Let (B, F_B) be the reduced pair, which is not hard, then every F_B-dicolouring of B, together with α, extends to an F-dicolouring of D. Then, in linear time, we either find an F_B-dicolouring of B, or we compute a valid pair $(B, \tilde{F}_B = (\tilde{f}_1, \tilde{f}_2))$ such that $(B, \tilde{F})$ is not hard, and an F_B-dicolouring of B can be computed in linear time from any $\tilde{F}_B$-dicolouring of B. This is possible by Lemma 5.3.10.

If we have not yet found an F_B-dicolouring of B, we compute an $\tilde{F}_B$-dicolouring of B in linear time, which is possible by Lemma 5.3.20. From this we compute an F_B-dicolouring of B in linear time. As mentioned before, this F_B-dicolouring together with α gives an F-dicolouring of D, obtained in linear time. □

Part II

(Dis)connecting digraphs

Chapter 6

Introduction: Altering the connectivity of (di)graphs

This second part of the thesis deals with algorithmic connectivity problems in digraphs. Specifically, we look at local modifications rendering a digraph either acyclic, or strongly connected. In Section 6.1, we are concerned with the connectivity augmentation of (di)graphs, and its restrictions, towards the plane strong connectivity augmentation problem studied in Chapter 7. In Section 6.2, we look at rendering a (di)graph acyclic through vertex deletions, arc deletions and the more general operation of inversions, which is the topic of Chapter 8.

6.1 Connectivity augmentation

Augmentation problems broadly ask for the addition of few edges ensuring the graph satisfies a given property. These questions have important applications in the field of survivable network design. Central to those are connectivity augmentation problems, asking for a minimum number of edge (arc) additions such that the resulting (di)graph attains a certain connectivity. These problems admit many variants, where unrestricted versions often admit polynomial algorithms, but various restrictions render the problem hard. Here, we will focus more specifically on restrictions asking for the resulting (augmented) graph to be planar.

To understand the generality of connectivity augmentation, let us observe that many well-known algorithmic problems can be rephrased in this language. Computing a shortest (weighted) path between two vertices of G consists in asking for a minimum-cost augmentation of an empty graph, allowing only for edges of G, such that the two vertices are connected. The minimum spanning tree problem can also be formulated similarly, and both problems admit polynomial-time algorithms. In-between these, specifying a subset of vertices (terminals) to connect at minimum cost yields the Steiner Tree problem, which is yet another one of Karp's NP-hard problem list [Kar72].

In essence, connectivity augmentation problems ask to augment a non-empty graph, and become interesting in their own sake when asking for connectivity at least 2. As we will see, edge-connectivity augmentation problems are sensibly simpler than vertex-connectivity ones. This is in particular because they have often been considered when allowing parallel edges (or arcs), which is of no help to obtain vertex-disjoint paths.

We begin this section with Subsection 6.1.1, exploring undirected connectivity augmentations, before considering their planar restrictions in Subsection 6.1.2. We then move to directed connectivity augmentation problems in Subsection 6.1.3. Constrained versions of these problems have been less studied than their undirected variants, and mostly deal with strong connectivity augmentation. We survey various restrictions of strong connectivity augmentation in Subsection 6.1.4. Finally, we introduce our problems of planar strong connectivity augmentation in Subsection 6.1.5, towards Chapter 7.

6.1.1 General connectivity augmentation

In the following, we investigate connectivity augmentations of undirected graphs. There, we distinguish two variants, either asking to increase the edge-connectivity, or the vertex-connectivity. For an extensive survey on connectivity augmentation, we refer the reader to the chapter by Frank and Jordan [FJ15].

The first general (edge) connectivity augmentation problem was introduced by Frank and Chou [FC70]. They consider non-uniform connectivity requirements between each pair of vertices, and ask for a minimum augmentation satisfying those. Still, most research has considered uniform connectivity requirements, asking for the resulting graph to be c-(edge)-connected.

Unweighted case

We define the following unweighted version edge-connectivity augmentation, where the addition of parallel edges is allowed (meaning this is in fact a multigraph problem).

c-Edge Connectivity Augmentation (c-ECA) **Input:** A graph G **Question:** Find a minimum multiset $X \subseteq \binom{V}{2}$ such that $G + X$ is c-edge-connected.

Cai and Sun [CS89] gave a combinatorial characterization of the minimum number of edges required to c-edge connect a graph. Building on their result, an efficient algorithm was obtained for the c-Edge Connectivity Augmentation problem by Watanabe and Namakura [WN87].

Theorem 6.1.1 (Watanabe, Nakamura '87). *c-ECA admits a polynomial-time algorithm.*

Imposing vertex-disjointness conditions renders parallel edges useless, which motivates the following definition for vertex connectivity augmentation. When clear from the context, this will be the default connectivity augmentation and we may drop the term "vertex".

c-(VERTEX) CONNECTIVITY AUGMENTATION (c-VCA)
Input: A graph G
Question: Find a minimum subset $X \subseteq \binom{V}{2}$ such that $G + X$ is c-vertex-connected.

The c-VCA problem seems considerably harder, with polynomial algorithms known only for $c = 2$ due to Eswaran and Tarjan [ET76], and $k = 3$ due to Watanabe and Nakamura [WN90]. Determining whether c-VCA admits a polynomial-time algorithm is one of the most pressing problems in the field. Still, it was shown to admit an FPT algorithm by Jackson and Jordan [JJ05].

Theorem 6.1.2 (Jackson, Jordan '05). *c-VCA is FPT.*

An interesting subcase of this problem asks to augment the connectivity by exactly one. Jordan [Jor95] first considered this problem and gave an approximation, then Végh provided a (uniform) polynomial-time algorithm [Vég11].

Theorem 6.1.3 (Végh '11). *There exists a polynomial-time algorithm solving c-VCA when the input graph is $(c-1)$-connected.*

Weighted variants

What if we now ask for augmentations where not all edges are available for the augmentation? Even more general, we can introduce a weight on each edge, and ask for solutions of minimum cost, yielding WEIGHTED c-VCA and WEIGHTED c-ECA . In this case, both connectivity augmentations become NP-hard even for $k = 2$, as shown by Eswaran and Tarjan [ET76].

Theorem 6.1.4 (Eswaran, Tarjan '76). *Both WEIGHTED 2-ECA and WEIGHTED 2-VCA are NP-complete.*

Indeed, both problems can be reduced directly from HAMILTONIAN CYCLE. A graph G is Hamiltonian if and only if the empty graph on $V(G)$ vertices can be augmented at cost $|V(G)|$, by assigning weights 1 to $E(G)$, and 2 to the rest. In fact, the edge version is even hard for trees, as shown by Frederickson [FJ81]. Furthermore, WEIGHTED 2-ECA is APX-hard already when restricted to trees, but admits a $1.5 + \epsilon$-approximation (for any $\epsilon > 0$) due to a recent result by Traub and Zenklusen [TZ23].

When it comes to fixed-parameter tractability, we still lack a general FPT algorithm. Still, for the specific case of increasing edge-connectivity by one, a polynomial kernel was obtained by Marx and Végh [MV15].

Theorem 6.1.5 (Marx, Végh '15). *WEIGHTED c-ECA admits a polynomial kernel on $(c-1)$-edge-connected graphs.*

In another direction, Frank [Fra92] considered a vertex-weighted version of edge-connectivity augmentation, which he showed to be polynomial-time solvable.

Edge-connectivity augmentation of simple graphs

The c-ECA problem allows the addition of parallel edges, making it rather a multigraph problem than one about graphs. We look here at an important structural restriction on the output graph: simplicity. The problem of determining the complexity of this problem was already raised by Frank [Fra94]. This was settled by Jordan [Jor97], who showed that this restriction already makes for a hard problem.

Theorem 6.1.6 (Jordan '97)**.** *Simple c-ECA is NP-hard.*

On the positive side, Bang-Jensen and Jordan obtained the following result [BJ98]

Theorem 6.1.7 (Bang-Jensen, Jordan '98)**.** *Simple c-ECA is FPT.*

6.1.2 Planar connectivity augmentations

We have seen that restrictions on the set of allowed edges (through weights), as well as simplicity requirements, turn connectivity augmentation into a harder problem. Asking for simplicity is a first way to impose a structural requirement on the resulting graph. Another natural such restriction, and probably the most studied one, is to ask for the augmented graph to be planar. This restriction will be our main interest in this thesis, but other conditions have been studied, such as requiring bounded maximal degree [Mas+05]. Outside the case of multigraphs, planarity implies the existence of a vertex with degree at most 5, meaning we should only ask connectivity increases of low values. In the following, we survey various undirected planar augmentation problems, starting with the well-studied subcase of Steiner tree, before considering more general connectivity augmentations constrained by planarity.

Planar Steiner problems

Recall the Steiner Tree problem can be seen as a weighted 1-connectivity augmentation problem over the terminals. It is one of the most studied problems in parameterized complexity, making for a good test case before asking questions about connectivity. When parameterized by the number of terminals k, the problem admits an FPT $O(3^k \cdot n^{O(1)})$ algorithm, by a result Dreyfus and Wagner [DW71]. When the input is planar, this is the Planar Steiner Tree problem, and we could ask whether this structural guarantee allows for more efficient algorithms. Assuming the exponential-time hypothesis, this possibility was discarded by Marx and Pilipczuk [MPP18].

Theorem 6.1.8 (Marx, Pilipczuk, 18)**.** *Unless the ETH fails, there is no $O(2^{o(k)}n^{O(1)})$ algorithm for Planar Steiner Tree.*

As the authors note, this makes it one of the first planar problems not witnessing the "square-root phenomenon", which describes the fact that many problems admit $O(2^{O(\sqrt{k})}n^{O(1)})$ algorithms in the planar setting. Approximation-wise, planarity can be obtained to improve on the known 2-approximation, as a PTAS was obtained by Borradaille et al. [BKM09].

Simple circuit

The first example of a question which can be considered as a connectivity augmentation problem constrained by planarity was introduced by Rappaport in 1989. In [Rap89], he investigated the problem of augmenting a set of lines in the plane into a polygon.

> Simple Circuit
> **Input:** A set of embedded line segments S
> **Question:** Can S be augmented to a simple circuit?

The above can be seen as a generalization of the famous travelling salesman problem. In (Euclidian) TSP, we consider (embedded) independent vertices and ask for a minimum length circuit passing through all. Without weights, the problem is trivial, but imposing some of the edges of the circuit yields the Simple Circuit problem. The resulting graph (polygon) being plane is important, as augmenting non-embedded lines while preserving planarity is trivial. As Rappaport showed, this restriction is enough to make for a hard problem [Rap89].

Theorem 6.1.9 (Rappaport '89)**.** *Simple Circuit is NP-complete.*

On the positive side, he showed that convexity restrictions on the segments allow for polynomial algorithms [RIT90].

Plane and planar connectivity augmentation

Simple circuit is in fact a specific case of augmenting a plane straight-line graph to a 2-connected one. This can be naturally be generalized in two ways, by lifting some of the embedding constrains. First, we may consider the following *plane* problem, which still fixes an embedding but not as straight-line segments.

> Plane c-(Edge)-Connectivity Augmentation
> **Input:** A plane graph G, given with its embedding
> **Question:** Find a minimum set $X \subseteq \binom{V}{2}$ such that $G + X$ is c-(edge)-connected and plane, for the same embedding.

Then, the *planar* version is the same problem except it omits the input of an embedding to be preserved.

> Planar c-(Edge)-Connectivity Augmentation
> **Input:** A planar graph G
> **Question:** Find a minimum set $X \subseteq \binom{V}{2}$ such that $G + X$ is c-(edge)-connected and planar.

In these problems, we implicitly forbid parallel edges, meaning these questions are only non-trivial when $k \leq 5$. The most studied case considers $k = 2$, and was initiated by Kant and Bodlaender [KB91], who showed planar biconnectivity augmentation is hard.

Theorem 6.1.10 (Kant, Bodlaender '92)**.** *Planar 2-VCA is NP-complete.*

Still, in [KB91], the authors give a 2-approximation algorithm, and a 5/4 approximation for 3-VCA if the input is 2-connected. For $k = 2$, Gutwenger, Mutzel and Zey [GMZ09] later improved the approximation factor to 5/3. Moreover, if the input graph is outerplanar, Planar 2-VCA admits a linear-time algorithm for every c due to Kant [Kan96].

Hardness for the edge variant was obtained much later, by Rutter and Wolff [RW08].

Theorem 6.1.11 (Rutter, Wolff '08)**.** *Planar 2-ECA is NP-complete.*

Again, if the input graph is outerplanar, then Planar c-ECA is polynomial-time solvable, according to Nagamochi and Eades [NE98].

Regarding plane connectivity augmentations, the authors of [GMZ09] showed that the vertex-connectivity version is NP-hard in general.

Theorem 6.1.12 (Gutwenger, Mutzel, Zey '09)**.** *Plane 2-VCA is NP-complete.*

On the other hand, if the input is connected, the problem becomes tractable.

Theorem 6.1.13 (Gutwenger, Mutzel, Zey '09)**.** *Plane 2-VCA admits a polynomial-time algorithm for connected graphs.*

Similar results have been obtained for connectivity augmentation of plane straight-line graphs by Akitaya et al. [Aki+19].

To the best of our knowledge, parameterized aspects of these problems have not yet been considered, and we ask the following.

Question 6.1.14. *Do planar (plane) connectivity augmentation problems admit FPT algorithms when parameterized by the solution size?*

Let us note that planarity-preserving augmentations have been considered for other requirements than connectivity. For example, one can ask for the resulting graph to have low diameter [GRT15], or to be Hamiltonian [DL10] We refer to the survey of Hurtado and Toth [HT13] for an overview of the field.

6.1.3 Directed connectivity augmentations

The analogue of connectivity augmentation for digraphs is to ask for arc additions rendering a digraph c-strongly connected or c-arc-strongly connected. We will take particular interest in the case $c = 1$, that is, Strong Connectivity Augmentation, towards structurally constrained variants explored later. Note that when $c = 1$, augmenting with multiple copies of the arc is of no interest, meaning the only "simplicity" consideration is whether we allow digons or not.

Unrestricted case

Without restrictions, asking for a minimum set of arcs to make a digraph D strongly connected is not a hard task. If D has source components S and sink components T, at least $\max(|S|, |T|)$ arcs must be added incident to them, giving a lower bound on the solution size. As shown by Eswaran and Tarjan [ET76], this is always sufficient, and a solution can be constructed efficiently.

Theorem 6.1.15 (Eswaran, Tarjan '76)**.** *Given a digraph D, a minimum strongly connected augmentation of D can be computed in polynomial-time.*

Indeed, this problem easily reduces to DAGs by considering the condensation of D. Then, the algorithm consists in (carefully) adding arcs from sinks to sources "chaining" them into a directed circuit using $\min(|S|, |T|)$ arcs, then adding arcs between the resulting SCC and the remaining terminals.

The multidigraph version of c-arc strong connectivity can also be solved efficiently, by a result of Frank [Fra92].

Theorem 6.1.16 (Frank '92)**.** *c-Arc Strong Connectivity Augmentation for multidigraphs is polynomial-time solvable.*

Later, Frank and Jordan [FJ95] showed even the c-strongly connected version is FPT, for simple digraphs, and a polynomial-time algorithm was given by Végh and Benczur [VB08].

Theorem 6.1.17 (Végh, Benczur '08)**.** *c-Strong Connectivity Augmentation is polynomial-time solvable.*

Weighted SCA

In light of this, the general Strongly Connectivity Augmentation refers to the weighted version of the problem, defined as follows.

> (Weighted) Strong Connectivity Augmentation (SCA)
> **Parameter:** k
> **Input:** A digraph $D = (V, A)$, a set of links $L \subseteq V^2$, a weight function $w : L \to \mathbb{R}^+$, an integer k and a budget $\alpha \in \mathbb{R}^+$.
> **Question:** Is there a subset $X \subseteq L$ such that $D + X$ is strongly connected, $|X| \leq k$ and $\sum_{e \in X} w(e) \leq \alpha$?

Like its undirected 2-connectivity counterparts, this problem admits a straightforward reduction from the NP-hard (directed) Hamiltonian cycle problem [Kar72].

Theorem 6.1.18 (Eswaran, Tarjan '76)**.** *(Weighted) Strong Connectivity Augmentation is NP-complete.*

A 2-approximation algorithm was obtained by Frederickson and Ja'Ja [FJ81] already in 1981. The fixed-parameter tractability of this problem was only established recently by Klinkby, Misra and Saurabh [KMS21].

Theorem 6.1.19 (Klinkby et al. '21)**.** *(Weighted) Strong Connectivity Augmentation is fixed-parameter tractable, and admits a $2^{O(k \log k)} \cdot n^{O(1)}$ algorithm.*

Their technique consists in reducing the problem to *graph pattern recognition*, where one tests for all strongly connected "patterns" on $3k$ vertices, and tries to map those onto augmentations of the instance of SCA.

6.1.4 Constrained Strong Connectivity Augmentation

Allowing only digons

A first interesting "structural" restriction is to ask for a strongly connected augmentation of D by only turning existing arcs into digons. That is, we want a minimum X such that, letting $\overleftarrow{X} = \{(y, x) : (x, y) \in X\}$, the digraph $D + \overleftarrow{X}$ is strongly connected. Recalling Subsection 1.2.5, it is not hard to see that the above is equivalent to asking for a minimum dijoin of D. Indeed, adding all the reverse arcs of a dijoin cancels all dicuts of D, resulting in a strongly connected digraph. The following fundamental min-max theorem due to Lucchesi and Younger [LY78] expresses a strong duality between dijoins and dicuts.

Theorem 6.1.20 (Lucchesi, Younger '78)**.** *The minimum size of a dijoin equals the maximum number of disjoint dicuts.*

In fact, this is a consequence of a much stronger result due to Edmonds and Giles [EG77] concerning submodular functions. Frank provided an algorithmic proof of both of these results [Fra81, Fra82], see also [Fra84].

Theorem 6.1.21 (Frank '81)**.** *There exists a polynomial-time algorithm computing a minimum dijoin of a digraph.*

The theorems of Frank are much more general, and in particular apply to the weighted version of our problem. Still, we will only need the algorithm above for our considerations.

Bipartition constraints

The first truly structural restriction of strong connectivity augmentation comes from the grid bracing problem. In terms of digraphs, this problem can be formulated by asking for a bipartition-preserving strong connectivity augmentation.

> BIPARTITE STRONG CONNECTIVITY AUGMENTATION
> **Input:** A bipartite graph G
> **Question:** Find a minimum subset $X \subseteq V(D)^2$ such that $D + X$ is strongly connected and bipartite.

This problem was shown to be polynomial-time solvable by Gabow and Jordan [GJ00], who also obtained similar results when preserving partitions into a larger number of classes.

Theorem 6.1.22 (Gabow, Jordan '00). *BIPARTITE STRONG CONNECTIVITY AUGMENTATION is polynomial-time solvable.*

Planar Directed Steiner problems

Here, we explore the (hard) parameterized complexity of Steiner-type problems for planar digraphs. The canonical generalization of STEINER TREE to digraphs asks for an arborescence of minimum weight attaining the terminals. It also admits generalizations asking for the resulting "steiner subgraph" to be strongly connected, and these problems are well-studied in parameterized complexity. In the STRONGLY CONNECTED STEINER SUBGRAPH (SCSS) problem, we are given a digraph D along with terminals T, and ask for a strongly connected subgraph of minimum weight containing T. In the general case, this problem is NP-hard, and even $W[1]$-hard parameterized by $k = |T|$ by a result of Guo et al. [GNS11]. A finer version of this problem is the DIRECTED STEINER NETWORK (DSN) problem, prescribing reachabilities between the terminals. We are specifically interested in the planar versions of these problems. There, while more efficient algorithms can be attained for the first problem, Chitnis et al. [Chi+20] have shown both problems to be among the few $W[1]$-hard planar problems.

Theorem 6.1.23 (Chitnis et al. '19). *PLANAR DSN and PLANAR SCSS are $W[1]$-hard parameterized by the number of terminals.*

Let us note that the parameterized complexity of PLANAR DSN was fully classified by Galby et al. [Gal+24], depending on the prescribed reachabilities.

6.1.5 Plane strong connectivity augmentation

In the following, we introduce the contributions of Chapter 7 on strong connectivity augmentation of plane oriented graphs. We stress that only plane versions of the problem are considered, that is, the input digraph is given with an embedding which must be preserved by the augmentation. Then, there are two ways to define the problem. The first version considers all planar digraphs as input and allow for the addition of digons.

> k-DIRECTED-PSCA (DPSCA)
> **Parameter:** k
> **Input:** A plane digraph D, given with its embedding
> **Question:** Is there a set of arcs $X \in V(D)^2$ of size at most k such that $D + X$ is plane and strong, while preserving the embedding of D?

Our main consideration in this thesis forbids digons, and can be seen as a simplicity-preserving version of the problem, considering only oriented graphs.

k-PLANE STRONG CONNECTIVITY AUGMENTATION (PSCA) **Parameter:** k **Input:** A connected oriented plane graph D **Question:** Is there a set of arcs $X \in V(D)^2$ of size at most k such that $D + X$ is strong and plane, with the same induced embedding for D ?

Observe that a plane digraph can always be augmented to strong connectivity by adding any maximal set of arcs (yielding a bidirected triangulation). This is not the case for oriented graphs, as witnessed by the digraph consisting of a single arc. This means that even the question of augmentability for plane oriented graphs is of interest. Our first result is that this is indeed not a trivial question.

Theorem 6.1.24 (Bessy, Gonçalves, R., Thilikos '25)**.** *PLANE STRONG CONNECTIVITY AUGMENTATION is NP-complete, even without restrictions on the solution size.*

Our instances are 3-connected, which also yields NP-hardness of the PLANAR SCA problem. Although PLANAR 2-VCA and PLANAR 2-ECA are NP-hard, the hardness of PLANE SCA stands in stark contrast with the undirected analogue for 2-connectivity. Indeed, while PLANE 2-VCA is hard for disconnected instances, it admits a polynomial-time algorithm otherwise by Theorem 6.1.13.

Our main result is that PLANE STRONG CONNECTIVITY AUGMENTATION admits a fixed-parameter tractable algorithm.

Theorem 6.1.25 (Bessy, Gonçalves, R., Thilikos '25)**.** *PLANE STRONG CONNECTIVITY AUGMENTATION is FPT, and admits a $2^{O(k \log^2 k)} \cdot n^{O(1)}$ algorithm.*

For the directed version, where hardness is not known, we provide an FPT algorithm parameterized by the number of terminals, which is lower than $2k$ for positive instances.

Theorem 6.1.26 (Bessy, Gonçalves, R., Thilikos '25)**.** *k-DIRECTED-PSCA is FPT with respect to the number of terminals τ, and admits a $2^{O(\tau)} \cdot n^{O(1)}$ algorithm.*

6.2 Rendering a (di)graph acyclic

In this section, we deal with the problem of disconnecting a (di)graph by a minimum number of vertex and edge (arc) deletions. The first way to ask this question is to understand it globally: how to disconnect a graph G? How to make a digraph D not strongly-connected? The answers to these are given by separators and cuts, as we saw in Subsection 1.1.7. Here, we are interested in disconnections in a more global sense, through feedback problems, decreasing the connectivity between every pair of vertices. We formalize this question in Subsection 6.2.1, introducing feedback vertex sets and feedback arc sets. We survey algorithmic results about the computation of feedback vertex sets in Subsection 6.2.2. Then, we look at the feedback arc set problem in Subsection 6.2.3, which will be of particular interest to us, and is in a sense equivalent to feedback vertex set in digraphs. We further investigate feedback arc set for restricted classes of digraphs, and in particular tournaments, in Subsection 6.2.4. The topic of Chapter 8 is inversions, which are operations generalizing feedback arc sets, consisting in reversing all arcs within an induced subdigraph. We introduce algorithmic and structural problems about inversions in Subsection 6.2.5. Then, in Subsection 6.2.6, we consider the restriction of inversions to sets of bounded size, presenting our first algorithmic results. Finally, we consider inversions of sets with prescribed (fixed) size in Subsection 6.2.7, where we present upper bounds on the "inversion number", as well as hardness results.

6.2.1 Towards feedback problems

We have already seen in Subsection 1.1.7 that asking for any pair of vertices to be disconnected is trivial when considering edge deletions, and corresponds to Vertex Cover for vertex deletions. A natural way to relax this full disconnection is by allowing each pair of vertices to be connected, but not 2 (vertex or edge)-connected. By the theorems of Menger, for any pair $x, y \in V$ in the resulting graph, there cannot be two vertex-(edge-)disjoint paths between x and y. Observe then that this is equivalent to asking for the resulting graph to be a forest. This motivates the following definitions. In a graph G, a *feedback vertex set* is a subset $X \subseteq V$ such that $G - X$ is a forest. Similarly, a *feedback edge set*, is a subset $F \subseteq E$ such that $G - F$ is a forest.

In digraphs, if the goal is to disconnect all pairs of vertices, the story is the same as for graphs. The first interesting case, allowing *some* connectivity, was already posed in Question 1.2.3. That is, we allow pairs x, y in the resulting graph to admit either a path from x to y or a path from y to x, but not both. This is saying that the resulting digraph should be acyclic. Symmetrically to the above, considering a digraph D, a (directed) *feedback vertex set* is a subset $X \subseteq V$ such that $D - X$ is a DAG. Then, a *feedback arc set*, is a subset $F \subseteq E$ such that $D - F$ is acyclic. We denote the minimum size of a feedback arc set by $\text{fas}(D)$. Here, there is an equivalence between these two objects (see 10.3.1 in [BG02]). If we consider the line digraph of a given digraph D, feedback vertex sets correspond exactly to feedback arc sets of D, and vertex-disjoint cycles to edge-disjoint cycles of D. Conversely, if we split each $v \in V(D)$ into an arc (v^-, v^+), where all in-(out-)neighbours of v are affected to v^- (v^+), feedback arc sets and arc-disjoint cycles correspond to feedback vertex sets and vertex-disjoint cycles of D.

We will be interested in the computation of minimum feedback sets, which we will survey both for graphs and digraphs. We will see that, except for feedback edge set, all other variants are NP-complete, prompting us for parameterized algorithms and (parameterized) approximations.

Structure

What can be known about the structure of (di)graphs having no small feedback sets? Feedback sets look at breaking, or *hitting*, all (directed) cycles of a graph. Therefore, a collection, or *packing* of k (vertex or arc) disjoint cycles, is an obstacle to having a feedback set of size k, since one vertex must be taken in each of its cycles. Is there some link between this lower bound and the size of the feedback set? This question falls in the scope of many dualities between minimum hitting sets versus maximum packing of objects. In fact, we have already seen an example of this in a very strong form for local connectivity. Indeed, obstacles to small separators (cuts) between $x, y \in V$ are exactly what separators want to hit: vertex (edge) disjoint paths between x and y. There, Menger's theorem gives equality between the minimum hitting set (separators and cuts) and the maximum packing (vertex or edge disjoint paths).

For feedback vertex sets, this link was obtained in the following celebrated theorem by Erdős and Pósa [EP65].

Theorem 6.2.1 (Erdös, Pósa '65)**.** *For every graph G, either G contains k vertex-disjoint cycles, or it admits a feedback vertex set of size $O(k \log(k))$.*

In 1973, Younger made the analogue conjecture to the Erdös-Pósa theorem [You73], which was proved by Reed et al. in 1996 [Ree+96].

Theorem 6.2.2 (Reed, Robertson, Seymour, Thomas '96)**.** *There exists a function f such that for every digraph D, either D contains k vertex-disjoint circuits, or it admits a directed feedback vertex set of size $f(k)$.*

The analogous result is known for edges, either G admits k edge-disjoint cycles, or it has a feedback edge set of size $2k \log(k)$ (exercise 9.5 in [Die17]). It is also known for feedback arc sets and arc-disjoint directed cycles as a consequence of Theorem 6.2.2, because of the reduction to the vertex version in the line digraph.

6.2.2 Feedback vertex set

In this subsection, we survey results around the Feedback Vertex Set (FVS) problem in graphs and digraphs. Given a (di)graph G, the decision variant of (Directed) FVS asks, for a given k, whether there exists a feedback vertex set of size at most k. The minimization variant asks for one of minimum size. Historically, Feedback Vertex Set and its directed version were among the first problems shown to be NP-complete by Karp already in his seminal 1972 paper [Kar72].

The undirected problem admits a 2-approximation thanks to Becker and Geiger [BG96], but is APX-complete by a reduction from Vertex Cover (see [AK00]). The directed variant admits an approximation up to a factor dependent on the optimum, by a result of Seymour [Sey95]. Still, under the unique games conjecture there exist no constant-factor approximation, see Guruswami and Lee [GL16].

Parameterized algorithms

We now consider the parameterization of the decision variant by solution size, which we formally state below.

Feedback Vertex Set (k-FVS)
Parameter: k
Input: A graph G and an integer k
Question: Is there a set $X \subseteq V(G)$ of at most k vertices such that $G - X$ is a forest?

Directed Feedback Vertex Set (k-DFVS)
Parameter: k
Input: A digraph D and an integer k
Question: Is there a set $X \subseteq V(G)$ of at most k vertices such that $G - X$ is acyclic?

In the undirected case, the fixed-parameter tractability of Feedback Vertex Set was first shown by Bodlaender [Bod94]. There, he also shows FPT algorithms for the dual problem deciding packings of at least k vertex-disjoint cycles, and the edge-disjoint version. Since then, Feedback Vertex Set has been among the most studied parameterized problems, with a series of improvements, culminating in a result by Kociumaka and Pilipczuk [KP14].

Theorem 6.2.3. *k-Feedback Vertex Set admits a $\left((2+\phi)^k \cdot n^{O(1)}\right)$-time algorithm.*

Moreover, it was shown to admit a polynomial kernel by Thomassé [Tho10].

Theorem 6.2.4 (Thomassé '10)**.** *k-Feedback Vertex Set admits a kernel on $4k^2$ vertices.*

The Directed Feedback Vertex Set problem was eventually also shown to be FPT by Chen et al. [Che+08].

Theorem 6.2.5 (Chen et al. '08)**.** *k-Directed Feedback Vertex Set admits a $O(4^k k! \cdot n^{O(1)})$-time algorithm.*

6.2.3 Feedback arc set

In this subsection, we look at the Feedback Arc Set problem, asking for one of minimum size, which is the first case of our later problem on inversions. First, we note that the corresponding undirected problem, finding a minimum feedback edge set, is much simpler. Indeed, observe its complement in G must be a forest, a maximal one is one that is spanning, and then minimizes the complement feedback edge set. This can also be adapted for the weighted question by using Prim's algorithm, in fact discovered by Jarník [Jar30].

In the directed case, we still have an equivalence between the minimum feedback arc set and the maximum acyclic subdigraph. This is of much less help here, and both problems are hard, as Feedback Arc Set was also among the first NP-complete problems from Karp's list [Kar72]. This hardness is a consequence of the reduction from FVS by considering the digraph obtained by splitting each vertex into an arc.

The converse the reduction from FAS to FVS obtained by taking the line digraph means exact algorithms for the latter can be used to solve FAS. This reduction does not preserve approximation ratios, where the best known approximation is in $O(\log(n)\log\log(n))$, far from the 2-approximation for FVS. We are interested mainly in the Feedback Arc Set problem parameterized by k, which looks for an exact solution, as stated below.

Feedback Arc Set (k-FAS)
Parameter: k
Input: A digraph D and an integer k
Question: Is there a set $X \subseteq A(G)$ of at most k arcs such that $D - X$ is acyclic?

In particular, the reduction to FVS yields the fixed-parameter tractability of FAS as a consequence of Theorem 6.2.5

Corollary 6.2.6 (Chen et al. '08)**.** *k-Feedback Arc Set admits a $O(4^k k! \cdot n^{O(1)})$-time algorithm.*

Bounds

On the non-algorithmic front, it is interesting to estimate how large solutions to Feedback Arc Set can get, relative to n. The *feedback-arc-set number* of a digraph D, denoted by $\mathrm{fas}(D)$, is the minimum size of a feedback arc set. Let $\mathrm{fas}(n)$ be the maximal value of $\mathrm{fas}(D)$ over all digraphs of order n, and note that it is attained by tournaments. Questions abound bounds for $\mathrm{fas}(n)$ were first raised by Erdös and Moon [EM65], who showed $\frac{1}{2}\binom{n}{2} \le \mathrm{fas}(n) \le \frac{1}{2}\binom{n}{2} + cn^{3/2}(\ln n)^{1/2}$. This was improved by Spencer [Spe71, Spe80], before Fernandez de la Vega [Fer83] achieved to show the following asymptotically tight estimation.

Theorem 6.2.7 (Fernandez de la Vega '83)**.** $\mathrm{fas}(n) = \frac{1}{2}\binom{n}{2} + \Theta(n^{3/2})$.

6.2.4 Feedback arc set in restricted classes

Considering the hardness of approximation in general digraphs, and because of practical applications, much research has studied restrictions to specific classes. We will mainly be interested in the case of tournaments, but another well-studied case are planar digraphs and their relaxations.

Planar digraphs

Unlike Plane Strong Connectivity Augmentation, the FAS problem is polynomial-time solvable in planar digraphs, and this is another consequence of the Theorem 6.1.20 by Lucchesi and Younger [LY78].

Theorem 6.2.8 (Lucchesi '78)**.** *Feedback Arc Set admits a polynomial-time algorithm on planar digraphs.*

It is not hard to show that the minimum feedback arc set of a plane digraph is equal to the minimum size of a dijoin, see Theorem 15.3.9 in [BG09] for a proof. This means that FAS reduces to finding a minimum dijoin in D, which is polynomial-time solvable from Theorem 6.1.21.

Tournaments

Asking for a feedback arc set in a tournament can be seen as asking for a ranking (ordering) of the vertices minimizing the number of "upsets" (backward arcs). The tournament restriction, Feedback Arc Set in Tournaments (or FAST), remains NP-complete as shown by Alon [Alo06] and Charbit, Thomassé, and Yeo [CTY07].

The most drastic difference here, is that FAST now admits a so-called polynomial-time approximation scheme, or *PTAS*. For a minimization problem (as FAST), a PTAS takes some fixed $\epsilon > 0$, and produces an algorithm running in polynomial time to output a $(1 + \epsilon)$-approximation. The following result is due to Mathieu and Schudy [KS07], and will be directly used in Chapter 8.

Theorem 6.2.9 (Mathieu, Schudy '07)**.** *For any $\epsilon > 0$, there exists a polynomial-time algorithm taking any tournament T outputting a $(1 + \epsilon)$-approximation of* $\mathrm{fas}(T)$.

This result also had consequences on the parameterized complexity side, where has been used by Bessy et al. to obtain a linear kernel [Bes+11].

Theorem 6.2.10 (Bessy et al. '11)**.** *For any $\epsilon > 0$, k-Feedback Arc Set in Tournaments admits a kernel of size $(2 + \epsilon)k$.*

Their kernel relies on reduction rules to output an equivalent instance of size subquadratic in k, then employs the PTAS of [KS07]. As for the PTAS, the polynomial kernelization algorithm has a time-complexity (largely) dependent on ϵ.

6.2.5 Inversions towards acyclicity

While Feedback Arc Set asks for *deletion* of arcs towards acyclicity, it can equivalently be stated in terms of arc *inversions*. Indeed, if F is a minimum feedback arc set of D if and only if the digraph obtained by reversing each arc of D is. To observe this, we can for instance look at an acyclic (topological) ordering given by $D - F$. Generalizing this operation to more than a single arc, in 2010, Belkhechine et al. [Bel+10] introduced the concept of inversion for sets of vertices. In an oriented graph $\overrightarrow{G}$, if X is a set of vertices of $\overrightarrow{G}$, the *inversion* of X consists in reversing the orientation of all arcs with both endvertices in X. In particular, the reversal of a single arc corresponds to the inversion of the set of its endvertices.

Analogously to feedback arc sets, the main object of interest here is the *inversion number* $\mathrm{inv}(\overrightarrow{G})$ of an oriented graph $\overrightarrow{G}$. This was first defined in [Bel+10] as the minimum number of inversions transforming $\overrightarrow{G}$ into an acyclic digraph. This raises plenty of structural and algorithmic problems, which have recently received a considerable attention. A first line of research, surveyed in Subsection 6.2.5, looks at obtaining lower and upper bounds on inversion numbers over (subclasses of) digraphs. The second direction, which we investigate in Subsection 6.2.5, studies the computation of $\mathrm{inv}(\overrightarrow{G})$ for a given $\overrightarrow{G}$. Since inversion number cannot increase when taking subgraphs, giving bounds over the class of all digraphs amounts to giving bounds for tournaments, which have been the most studied subclass since [Bel+10]. Computationally, questions have also mostly dealt with tournaments, as will be the case of our contributions.

Bounds

In terms of upper bounds, the first results look at estimating $\mathrm{inv}(n)$, defined as the maximum of $\mathrm{inv}(\overrightarrow{G})$ over all oriented graphs $\overrightarrow{G}$ of order n. Recall this is the same value when restricted to tournaments. Belkhechine et al. first showed that $\frac{n-1}{2} - \log(n) \leq \mathrm{inv}(n) \leq n - 3$ in [Bel+10]. Recently, Aubian et al. [Aub+25] and Alon et al. [Alo+24] independently improved these bounds to the following.

Theorem 6.2.11 (Aubian et al. '22, Alon et al. '24)**.** *For n sufficiently large, $n - 2\sqrt{n \log n} \leq \mathrm{inv}(n) \leq n - \lceil \log(n+1) \rceil$*

Since inversions preserve the same underlying graph, it is also natural to ask to transform a given orientation of a (labelled) graph into another (which is now not necessarily acyclic). Havet et al. [HHR24] introduced the *inversion diameter* for a labelled graph G, as the maximal number of inversions required to go between any two orientations of G. They show that inversion diameter is asymptotically tied to previously known variants of the chromatic number: acyclic chromatic number, star chromatic number, and oriented chromatic number.

Algorithmic problems

Since (for now) we have no constraints on the inversion size, no hardness result on FAS yields hardness of determining $\mathrm{inv}(\overrightarrow{G})$. Let us state the problem below, parameterized by solution size.

> INVERSION
> **Parameter:** k
> **Input:** An oriented graph $\overrightarrow{G}$ and an integer k
> **Question:** Is $\mathrm{inv}(G) \leq k$?

The first result about the complexity of this problem was in fact an XP algorithm for its restriction to tournaments. Observe inv cannot increase for (induced) subgraphs, so the tournaments T

satisfying $\mathrm{inv}(T) \leq k$ can be characterized as those forbidding minimal tournaments with inv $> k$. Belkhechine et al. [Bel+10] gave a non-constructive proof that the class of k-"critical" tournaments is bounded. For each fixed k, the presence of these can be checked in polynomial time, yielding a polynomial-time algorithm deciding whether inv $\leq k$. In contrast, Bang-Jensen et al. [BSH22] proved that for $k \leq 2$, k-Inversion is NP-complete, which was later generalized to all values of k by Alon et al. [Alo+24].

Theorem 6.2.12 (Alon et al. '24). *k-Inversion is NP-complete for every k.*

In the case of tournaments, they improve on the XP algorithm of [Bel+10], by showing fixed-parameter tractability.

Theorem 6.2.13 (Alon et al. '24). *k-Inversion admits an $O(f(k)n^2)$-time algorithm when restricted to tournaments.*

There is still no hardness result for Inversion in tournaments, which was conjectured by Bang-Jensen et al. [BSH22].

Conjecture 6.2.14 (Bang-Jensen et al. '22). *For every integer k, k-Inversion is NP-complete for tournaments.*

Briefly stepping in the opposite direction, we note Duron et al. [Dur+24] considered the minimum number of inversions to make a digraph k-(arc-)strong. There, they show lower and upper bounds, as well as hardness and inapproximability results for all values of k.

6.2.6 Inversions of bounded size

We are ready to consider inversions of bounded size, and introduce the algorithmic results of Chapter 8. To further tighten the link between feedback arc set (which inverts sets of size 2) and inversions (of unbounded size), it is natural to impose bounds on the sets being inverted. Yuster [Yus25] initiated the study of $(\leq p)$-*inversions*, which are inversions over sets of size at most p. We first look at his bounds, then investigate our algorithmic questions.

Bounds

Given an oriented graph $\overrightarrow{G}$, its $(\leq p)$-*inversion number*, denoted by $\mathrm{inv}^{\leq p}(\overrightarrow{G})$, is the minimum number of $(\leq p)$-inversions rendering $\overrightarrow{G}$ acyclic. Note that $\mathrm{inv}^{\leq 2}(\overrightarrow{G}) = \mathrm{fas}(\overrightarrow{G})$. Yuster studied the maximum $\mathrm{inv}^{\leq p}(n)$ of the $(\leq p)$-inversion numbers over all oriented graphs of order n. In fact, he only considers tournaments, but again those are the oriented graphs attaining the maximal bounds. From the asymptotically tight estimation of $\mathrm{fas}(n)$ in Theorem 6.2.7, we know that $\mathrm{inv}^{\leq 2}(n) = \frac{1}{2}\binom{n}{2} + \Theta(n^{3/2})$. For inversions of size 3, Yuster proved $\frac{1}{12}n^2 + o(n^2) \leq \mathrm{inv}^{\leq 3}(n) \leq \frac{257}{292}n^2 + o(n^2)$ and conjectured $\mathrm{inv}^{\leq 3}(n) = \frac{1}{12}n^2 + o(n^2)$. He also proved the following for larger values [Yus25].

Theorem 6.2.15 (Yuster '23). *For sufficiently large p, there exist $\epsilon_p > 0$ fand $\delta_p > 0$ such that:*

$$\left(\frac{1}{2p(p-1)} + \delta_p\right) n^2 + o(n^2) \leq \mathrm{inv}^{\leq p}(n) \leq \left(\frac{1}{2\lfloor p^2/2 \rfloor} - \epsilon_p\right) n^2 + o(n^2)$$

Algorithms

Of course, determining $\mathrm{inv}^{\leq 2}$ for digraphs and tournaments is NP-hard because FAS is. In Chapter 8, we consider the same question for $\mathrm{inv}^{\leq p}$, as follows, and show it is also hard within tournaments.

$(\leq p)$-INVERSION
Input: An oriented graph D and a positive integer k
Question: Does D admit a decycling $(\leq p)$-family of size at most k?

Theorem 6.2.16 (Bang-Jensen, Havet, Hoersch, R., Rambaud, Silva '25)**.** TOURNAMENT $(\leq p)$-INVERSION *is NP-complete, for any fixed integer p greater than* 2.

In light of the linear kernel given by Theorem 6.2.10 for FAS in tournaments, thus for $(= 2)$-INVERSION, it is natural to ask whether TOURNAMENT $(\leq p)$-INVERSION also admits (a polynomial) one, for any fixed p. We give a positive answer to the above in a stronger sense: the TOURNAMENT BOUNDED SIZE INVERSION problem, considering both p and k as inputs, admits a polynomial kernel in (the couple of parameters) p and k.

Theorem 6.2.17 (Bang-Jensen, Havet, Hoersch, R., Rambaud, Silva '25)**.** *For any* $\epsilon > 0$, TOURNAMENT BOUNDED SIZE INVERSION *admits a kernel of size* $(1 + \epsilon)k^2p^3 + o(k^2p^3)$.

In the negative, we show that BOUNDED SIZE INVERSION for general digraphs is $W[1]$-hard parameterized by p, even for $k = 1$.

6.2.7 Inversions of Prescribed size

Restricting the above, we introduce $(= p)$-*inversions*, which are inversions over sets of size exactly p.

$(= p)$-invertibility

A first question is whether an oriented graph is $(= p)$-invertible at all. That is, can it be made acyclic using (any number of) $(= p)$-inversions? Clearly, no non-acyclic oriented graph of order at most p is $(= p)$-invertible. But there are non $(= p)$-invertible oriented graphs of arbitrarily large size. For example, observe that a $(= 3)$-inversion does not change the parity of the out-degrees of the vertices of a tournament. Hence to be $(= 3)$-invertible, a tournament of order n must have as many vertices of even out-degree as TT_n, the transitive tournament of order n, that is $\lceil n/2 \rceil$. This is in fact a characterization, but what does this problem look like for larger values?

$(= p)$-INVERTIBILITY
Input: An oriented graph $\overrightarrow{G}$
Question: Is $\overrightarrow{G}$ $(= p)$-invertible?

We characterize (most) $(= p)$-invertible graphs, and use this to show the question is almost always simple, through the following.

Theorem 6.2.18 (Bang-Jensen, Havet, Hoersch, R., Rambaud, Silva '25)**.** $(= p)$-INVERTIBILITY *admits a polynomial-time algorithm for oriented graphs of order* $n \neq p + 1$.

Perhaps surprisingly, the case $n = p+1$ is indeed harder (except, as we will see, for tournaments).

Theorem 6.2.19 (Bang-Jensen, Havet, Hoersch, R., Rambaud, Silva '25). *Deciding whether an oriented graph of order n is $(n-1)$-invertible is NP-complete.*

Bounds

In the following, we consider the $(=p)$-*inversion number* $\text{inv}^{=p}(\overrightarrow{G})$ of an oriented graph $\overrightarrow{G}$, defined as the minimum number of $(=p)$-inversions rendering $\overrightarrow{G}$ acyclic, or $+\infty$ if $\overrightarrow{G}$ is not $(=p)$-invertible. Clearly $\text{inv}^{=p}(\overrightarrow{G}) \geq \text{inv}^{\leq p}(\overrightarrow{G})$ and $\text{inv}^{=2}(\overrightarrow{G}) = \text{inv}^{\leq 2}(\overrightarrow{G})$, but are these parameters related in the other direction? We investigate upper bounds on the $(=p)$-inversion number in terms of feedback-arc-set number or the $(\leq p)$-inversion number.

We first look at bounding $\text{inv}^{=p}$ in terms of p and fas. There, we identify a dichotomy between even and odd values of p, by showing no such bound can exist for odd p, but for even p, the following holds.

Theorem 6.2.20 (Bang-Jensen, Havet, Hoersch, R., Rambaud, Silva '25). *Let $p \geq 4$ be an even integer and let D be an oriented graph with at least $p+2$ vertices. Then $inv^{=p}(D) \leq (8p - 12)\frac{(\text{fas}(D)+1)}{2}$*

We show that this also implies a bound in terms of $\text{inv}^{\leq p}$, namely $inv^{=p}(D) \leq \binom{p}{2}(8p - 12)\,\text{inv}^{\leq p}(D)$. We improve on Theorem 6.2.20 for digraphs where fas is large, by showing $\text{inv}^{=p}(D) \leq 2\,\text{fas}(D) + (4p-9)n + 19/2$, again for even p. We also show the following subquadratic bound in n for $\text{inv}^{=p}(n) \leq \frac{n}{2\lceil p/2\rceil \cdot \lfloor p/2 \rfloor} + o(n^2)$.

Complexity

We will show the algorithmic problems considered for $\text{inv}^{\leq p}$ admit the same reductions towards $\text{inv}^{=p}$, yielding the following hardness result.

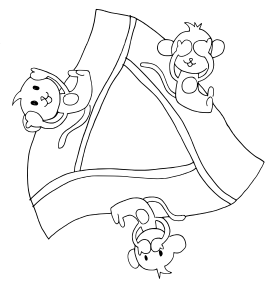
$(=p)$-INVERSION **Input:** A tournament D and a positive integer k **Question:** Does D admit a decycling $(\leq p)$-family of size at most k?

Theorem 6.2.21 (Bang-Jensen, Havet, Hoersch, R., Rambaud, Silva '25). TOURNAMENT $(=p)$-INVERSION *is NP-complete, for any fixed $p \geq 2$.*

Considering the problem PRESCRIBED SIZE INVERSION, where p is now part of the input, the same reduction as for BOUNDED SIZE INVERSION yields $W[1]$-hardness parameterized by p. An interesting question is whether it admits a polynomial kernel in (p, k) for tournaments, as in Theorem 6.2.17.

Chapter 7

Plane Strong Connectivity Augmentation



This chapter presents our contributions to strong connectivity augmentation within plane oriented graphs, following Section 6.1. It is based on joint work with Stéphane Bessy, Daniel Gonçalves and Dimitrios Thilikos, appearing in [Bes+25].

7.1 Introduction

In this chapter, we investigate plane strong connectivity augmentation within plane oriented graphs, through the k-PSCA problem which we recall.

> k-Plane Strong Connectivity Augmentation (k-PSCA)
> **Parameter:** k
> **Input:** A connected oriented plane graph D
> **Question:** Is there some $X \subseteq V(D)^2$ with $|X| \leq k$ such that $D + X$ is strong and plane, with the same induced embedding for D?

When referring to Plane Strong Connectivity Augmentation without mentioning k, the problem implicitly has no restrictions on k, and we simply ask whether there is any set X augmenting the graph while preserving planarity, without adding digons.

The restriction to oriented graphs rather than digraphs is motivated by *augmentability* considerations. Indeed, plane digraphs can always be augmented to strong connectivity while staying in the class of plane digraphs: any maximal augmentation yields a bidirected triangulation, which is strongly connected. On the other hand, within plane oriented graphs, augmentation to strong connectivity may not always be possible, take for example any oriented triangulation that is not already strongly connected. Our first result is that deciding strong connectivity augmentability within plane oriented graphs is already NP-hard.

Theorem 7.1.1. *Plane Strong Connectivity Augmentation is NP-complete.*

Our reduction is 3-connected, which also yields hardness of the Planar-SCA problem (without budget). These reflect the complexity of the (budgeted variants) in the undirected case (see Theorem 6.1.10 and Theorem 6.1.12). But the instances being 3-connected contrasts with the polynomial-time solvability Plane 2-VCA for connected instances. To our knowledge, Plane Strong Connectivity Augmentation is also one of the few connectivity augmentation problems to be NP-hard even without budget restrictions, where the only other example seems to be Simple Circuit.

Our main result is a fixed-parameter tractable algorithm for k-PSCA.

Theorem 7.1.2. *k-Plane Strong Connectivity Augmentation is FPT with respect to k, and admits a $2^{O(k \log k)} n^{O(1)}$ algorithm.*

In order to introduce the algorithm of Theorem 7.1.2, we will show a simpler procedure for the same question within directed graphs, defined below.

> k-Directed Plane Strong Connectivity Augmentation (k-Directed-PSCA)
> **Parameter:** k
> **Input:** A connected plane digraph D
> **Question:** Is there some $X \subseteq V(D)^2$ with $|X| \leq k$ such that $D + X$ is strong and plane, with the same induced embedding for D?

We show that k-Directed-PSCA is FPT parameterized by the number of terminals τ, which also implies its fixed-parameter tractability with respect to solution size.

Theorem 7.1.3. *k-Directed-PSCA is FPT with respect to the number of terminals τ, and admits a $2^{O(\tau)} n^{O(1)}$ algorithm.*

Scheme of the FPT algorithm(s). For both k-PSCA and k-Directed-PSCA, our strategy proceeds by searching for *structured* solutions. Intuitively, these correspond to solutions X where in each face F of D, the endpoints of X are skewed towards "terminal components" of the digraph induced by F. While such a notion admits a straightforward definition in k-Directed-PSCA, the case of k-PSCA is much more involved. Our main structural result is that the number of structured completions — possible restrictions of a structured solution to F — can be bounded by a function of k. Then, we distinguish simple and alternating faces, according to the number of "terminals" they induce. We show that for positive instances (D, k), the number of alternating faces is always bounded as a function of k (in fact, τ), which enables branching over all structured completions over them. This reduces our question to the computation of a solution adding arcs only within simple faces. The main obstacle then is that the number of simple faces can be linear in the size of the instance, but crucially, each of them admits an (absolute) constant number of "minimal" structured completions. For k-Directed-PSCA, this constant is always one, enabling a direct reduction to the problem of computing a minimal dijoin, which is polynomial by a result of Frank [Fra81]. This gives FPT algorithm parameterized by the number of terminals τ. In the case of k-PSCA, we use a randomized algorithm guessing one "allowed" completion per face, reducing our problem to Weighted SCA, which admits an FPT-algorithm due to Klinkby, Misra and Saurabh [KMS21]. This yields a FPT Monte-Carlo algorithm for k-PSCA, which we derandomize using universal sets.

Structure of the chapter. After some preliminaries in Section 7.2, we show the FPT algorithm for k-Directed-PSCA in Section 7.3, serving as an introduction to our k-PSCA algorithm. While the overarching steps are common, most technicalities of k-PSCA are short-circuited in k-Directed-PSCA, where structured solutions admit a very natural definition. Then Section 7.4 shows our FPT algorithm for k-PSCA, achieving to prove Theorem 7.1.2. Finally, we show the hardness of PSCA in Section 7.5, yielding Theorem 7.1.1.

7.2 Preliminaries

All digraphs in this chapter are plane and given with their embedding. We use the terms "strong" and "strongly connected" interchangeably, and often refer to directed paths as *dipaths*. For any D, the (possibly trivial) source and sink components of D form the set $\mathcal{T}(D)$ of *terminal components*, and we let $\tau(D) = |\mathcal{T}(D)|$.

Boundaries, labelled vertices, and angles. Given a plane digraph D, we identify a face F with its *boundary*, which is the (clockwise) closed walk following F. When D is not 3-connected, a given vertex or arc may appear multiple times along the boundary, which is why we specify each occurrence as a *labeled* vertex or an *angle*. An *angle* $\widehat{a_{i-1}v_i a_i}$ on a face F is the incidence of two consecutive edges a_{i-1} and a_i at v_i. We will always consider a labelled vertex v_i and its corresponding angle $\widehat{a_{i-1}v_i a_i}$ interchangeably. Then, we may simply refer to (subwalks of) a boundary by sequences of labelled vertices $(v_1, v_2, \ldots, v_\ell)$. Up to symmetry, we always consider those ordered clockwise.

Completions. Given a plane digraph D, a *completion* of D is some embedded subset $X \in V(D)^2$, such that $D{+}X = (V(D), A(D) \cup X)$ is plane and oriented (directed in Section 7.3), when embedded with respect to D and X. When $D{+}X$ is moreover strong, X is a *solution* to D. For any completion

X, and any face F of D, the *restriction* X_F of X to F is the subset of X embedded in F. Then, for a subwalk P of the boundary of F, the *restriction* X_P of X to P refers to the arcs of X_F with at least one endpoint on P. We say X is a completion of a face F if it is embedded in F.

7.3 FPT algorithm when allowing digons

The purpose of this section is to show Theorem 7.1.3, by describing a FPT algorithm for k-Directed-PSCA, when parameterized by the number of terminals τ. This directly yields a FPT algorithm parameterized by solution size k, thanks to the following observation.

Observation 7.3.1. *Any positive instance (D, k) for k-Directed-PSCA satisfies $|\tau(D)| \leq 2k$.*

Indeed, a solution must be incident to every terminal of D, and each arc can only be incident to two, meaning we can always output no when $k < \frac{1}{2}\tau(D)$.

Throughout the following, we consider an instance (D, k), where D is a plane *di*graph, and allow digons in the solution. We will therefore consider *completions* (and *solutions*) as embedded sets of arcs X of $V(D)^2$ such that $D + X$ is a plane *di*graph. We begin in Subsection 7.3.1 with a reduction to acyclic instances, the describe their structure: local terminals, as well as simple and alternating faces. We define structured solutions in Subsection 7.3.2, and show a solution can always be reconfigured into one that is structured. In Subsection 7.3.3, we bound the number of alternating faces, and local terminals within them, as a function of the number of terminals τ. This allows us to guess structured completions corresponding to restrictions of any possible structured solution to alternating faces in FPT time. Then, in Subsection 7.3.4, we show how to reduce the computation of the remaining solution in simple faces to the (polynomial) problem of finding a minimum dijoin, which yields our algorithm for k-Directed-PSCA.

7.3.1 Structure of the instances

Reducing to acyclic instances

We begin by reducing the problem to acyclic instances, through the following. For any plane digraph D, a *condensation* of D, D^c, is a plane acyclic digraph with loops and multiple arcs obtained by contracting iteratively every non-loop arc belonging to a strong component of D. Note that the embedding of D^c can differ according to the contraction sequence, but not the underlying digraph.

Lemma 7.3.2. *An instance (D, k) of Directed-PSCA admits a solution if and only if (D^c, k) does, for any condensation D^c. Moreover, any solution for D^c can be transformed into a solution for (D, k) in polynomial time.*

Proof. Let us consider a plane digraph D with possibly loops and multiple arcs, and let D' be the plane digraph obtained from D by contracting a non-loop arc (u, v) contained in a strong component C of D. Let w be the vertex of D' resulting from the merging of u and v. It suffices to show how to transform a solution X for D into a solution X' for D', with $|X| = |X'|$, and vice-versa.

Given a solution X for D, let us construct X' as follows: keep every arc not incident to u or v, and for the other arcs replace their endpoint u or v, by w. It is rather immediate that $D' + X'$ is plane and strong.

Given a solution X' for D', let us construct X as follows. Let us first distinguish some faces of D. Let $\widehat{u}_1, \widehat{u}_2, \widehat{v}_1, \widehat{v}_2$ be the angles at u or v incident to (u, v). Let $F_u(D), F_v(D)$ be the faces of D that are incident to u or v, respectively, at an angle distinct from $\widehat{u}_1, \widehat{u}_2, \widehat{v}_1, \widehat{v}_2$. Note that as the faces of D and D' have a natural bijection, $F_u(D), F_v(D)$ also denote sets of faces in D'. For constructing X, keep every arc that is not incident to w, and for the other arcs e, replace their endpoint w by u (resp. v), if e is embedded in $F_u(D)$ (resp. if e is not embedded in $F_u(D)$). The planarity of $D+X$ is clear, and for the strong connectivity, note that u and v are connected in both direction, in D, and thus also in $D+X$. Indeed, there is an arc from u to v, and as they belong both to C there is a dipath from v to u. This implies that any dipath going through w in $D' + X'$ can be rerouted through u and v. □

We may now reduce our problem to k-DAG-DIRECTED-PSCA, corresponding to k-DIRECTED-PSCA where the input is a plane DAG (without loops).

Lemma 7.3.3. *The problem k-DIRECTED-PSCA polynomially reduces to k-DAG-DIRECTED-PSCA.*

Proof. From Lemma 7.3.2, we know that k-DIRECTED-PSCA polynomially reduces to instances D that are DAGs with loops. We hence just have to deal with getting rid of the loops. Before going further, note that there is no point of having a loop in a solution X, so faces incident to only one vertex should not contain any arc of a minimum solution X.

For any loop ℓ of D based on a vertex v, let V_{in}, V_{out} be the sets of vertices embedded inside or outside ℓ, respectively, those sets being possibly empty. Clearly, any solution X for D divides into two solutions X_{in}, X_{out} (according to the position of the arcs with respect to ℓ) for the subgraphs $D_{in} = D[V_{in} \cup \{v\}]$ and $D_{out} = D[V_{out} \cup \{v\}]$. Indeed, any dipath with both ends in D_{in} (resp D_{out}) does not go through a vertex of V_{out} (resp. V_{in}). Conversely taking the union of two solutions X_{in}, X_{out} for D_{in} and D_{out}, leads to a solution for D. Indeed, this follows from the fact that every vertex of V_{in} (resp. V_{out}) has a dipath from v and a dipath towards v in D_{in} (resp. D_{out}).

Finally, observe that a completion X renders $D'_{in} = D_{in} - \{\ell\}$ (resp. $D'_{out} = D_{out} - \{\ell\}$) plane and strong if and only if it renders D_{in} (resp. D_{out}) plane and strong.

Hence, (D, k) is a positive instance if and only if there exists k_{in}, k_{out} such that $k = k_{in} + k_{out}$, and such that (D'_{in}, k_{in}) and (D'_{out}, k_{out}) are both positive instances. In order not to iterate on distinct values for the budgets (e.g. k_{in}, k_{out}), one should rather first note that (D, k) reduces into $O(n)$ non-trivial (i.e. with more than one single vertex) loop-free DAG instances D_i, for which we are only interested in the minimum value k_i for which they become positive. To find this minimum for D_i, we may just iterate over all values from 1 to $3|V(D_i)|$ (by density) with an algorithm solving DAG-DIRECTED-PSCA. Hence, one can solve the initial instance (D, k) by making $O(n^2)$ calls to this algorithm. □

Local terminals and face types

For a face F, the angle $\widehat{a_{i-1}v_ia_i}$, equivalently the labelled vertex v_i, is a *local terminal* if a_{i-1} and a_i are both leaving or both entering v_i. In the first case, v_i ($\widehat{a_{i-1}v_ia_i}$) is a *local source*, and in the other case it is a *local sink*. Then, observe that any subwalk from a local source to a (consecutive) local sink is necessarily a directed path, as any repeating vertex would create a cycle. Therefore, local sources and local sinks alternate along the boundary, joined by directed paths, and there are as many local sources as local sinks.

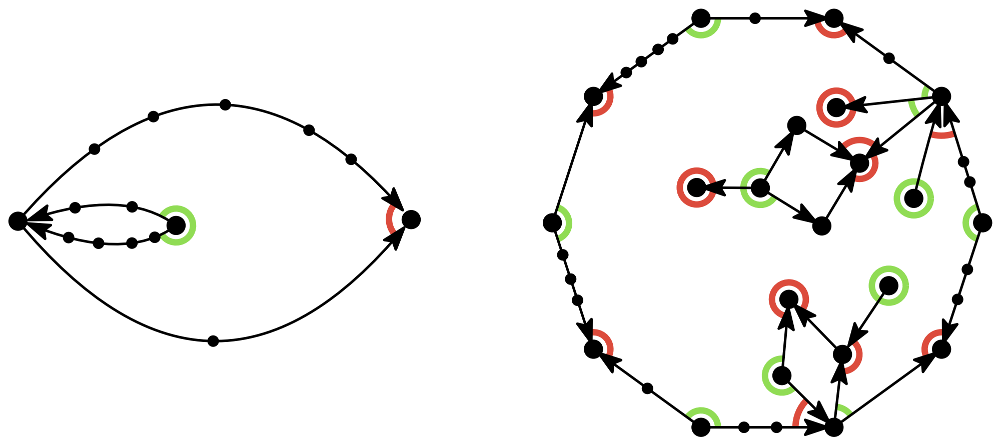


Figure 7.1: A simple face, with two local terminals and an alternating face with 22 local terminals. Local sources are shown as green angles, while local sinks are the red angles. Note that the same vertex can appear in both local sinks, local sources and even non-terminal angles.

We define $\mathtt{lt}(F)$ to be the number of local terminals of F, which is therefore even, and always non-zero when D is not reduced to a single vertex (in which case the instance is trivial). We say that a face F is a *simple* if $\mathtt{lt}(F) = 2$, and let $\mathcal{SF}$ be the set of simple faces of D. A face is *alternating* when $\mathtt{lt}(F) \geq 4$, and we let $\mathcal{AF}$ be the set of alternating faces of D. See Figure 7.1.

7.3.2 Existence of structured solutions

A completion, in particular a solution, X for an instance (D, k) of DAG-DIRECTED-PSCA is *structured* if every arc of X embedded in a face F lies between two local terminals of F. The following allows us to narrow our search only to structured solutions, which we will do by computing structured completions within each face.

Lemma 7.3.4. *If an instance of DAG-DIRECTED-PSCA admits a solution of size k, then it admits a structured solution of size k.*

Proof. Consider an instance (D, k), with D acyclic, and assume that it admits a completion X of size at most k. To prove the result, we fix an arbitrary face F of D and show how to reconfigure the restriction X_F into a structured solution while preserving its validity. If X_F is not structured, let P be any subwalk of the boundary of F from a local source s to a consecutive local sink t, such that X_F is incident to internal vertices of P. Recall that P is in fact a directed path, and we can consider its angles as (non-repeating) vertices. We let X_P be the subset of X_F incident to P, and describe how to successively reconfigure the endpoints of all its elements on P to either s or t.

First, if X_P contains any arcs with both endpoints on P, we modify X_P in X as follows. Remove all arcs with both endpoints on P and replace them with (t, s), then substitute all endpoints of X_P in P with s. It is easy to see that this can be done while preserving a planar embedding. The addition of (t, s) ensures that $V(P)$ is now strongly connected. Then, consider any arc (x, p) in the initial X_P, with $p \in P$ and $x \notin P$. Since its modification yields (x, s), note that the path from x

to p is preserved after the reconfiguration. Indeed, we may follow (x, s) then any path towards p in $V(P)$. This yields that the resulting completion is a solution, concluding the proof for this case.

Now, we may assume all arcs in X_P have exactly one endpoint in P. Take $v \in P \setminus \{s, t\}$ to be any (labelled) vertex incident to arcs of X_P, and let $X_v \subseteq X_P$ be the subset of those arcs. Our goal is to "shift" their v endpoint either towards s or t. Consider v, and let u (resp. w) be the previous (resp. next) vertex of P incident to X_P. When this is not defined, let $u = s$ (resp. $w = t$). We define X_u (resp. X_w) to be the subset of arcs obtained by substituting each endpoint of X_v on P with u (resp. with w). We will show that in X, X_v can be replaced with either X_u or X_w. After each such shift, arcs of X_P will be incident to exactly one less internal vertex of P, ensuring this operation ends.

Consider the digraph $D' = D + X - X_v$ and note that arcs of X_v form an oriented star joining every sink component to every source component in D'. We distinguish three cases according to the strong component of D' containing v. First, if v belongs to a source component S', the directed subpath $(s, \ldots, u \ldots, v)$ of P implies $u \in S'$ as well. In turn, $D' + X_u$ is strongly connected, because X_u still joins every sink component of D' to every source component in the same way as X_v. See Figure 7.2 for an illustration of this case. Symmetrically, if v belongs to a sink component T', we have $w \in T'$, which yields that $D + X_w$ is strongly connected. Otherwise, if v does not belong to a terminal component of D', substituting endpoint v with any vertex in P while preserving planarity also yields a solution and we may replace X_v by X_u or X_w arbitrarily.

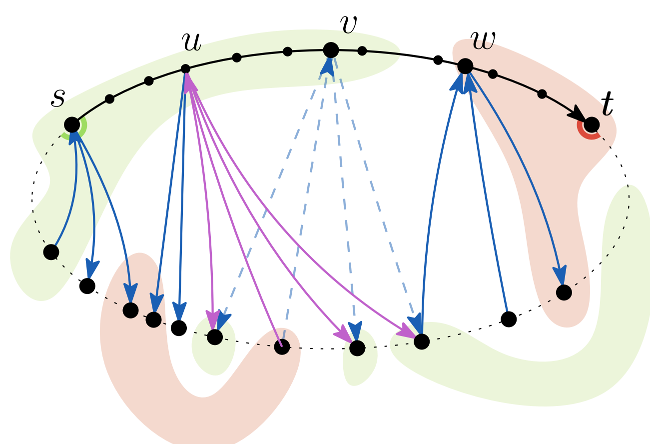


Figure 7.2: A directed path P from a local source s to a local sink t of a face, with arcs of a solution incident to it shown in blue. Removing the arcs incident to vertex v yields several source and sink components (in green and red), one of which is a source containing both u and v. Shifting their endpoints to u, as shown in purple, yields a solution incident to fewer internal vertices of P.

Therefore, either $X - X_v + X_u$ or $X - X_v + X_w$ is a solution for D which, by the choice of u and w, can be embedded planarly. Since the number of internal vertices of P incident to X_P strictly decreases, we can repeat this process until the only vertices of P incident to X_P are s and t. Applying this transformation for all possible choices of P in F yields a solution X in which X_F is structured. Repeating the process for each face of D, results in a structured solution. □

7.3.3 Branching over structured completions in alternating faces

In this subsection we show how to branch, in time FPT in τ, over all structured completions of alternating faces, corresponding to restrictions of a structured solution.

In order to bound the number of local terminals in alternating faces in terms of τ, we will need the following lemma, which can also be derived from the Poincaré index formula shown by Guattery and Miller [GM92].

Theorem 7.3.5 (2.1 in [GM92])**.** *For every plane DAG D, we have*

$$\sum_{F \in F(D)} (\mathtt{lt}(F) - 2) \leq 2\tau(D) - 4$$

Proof. Given a plane DAG D, and a vertex $v \in V(D)$, let $\overline{\mathtt{lt}}(v)$ denote the number of angles at v that are not local terminals. As there are $2|A(D)|$ angles in D we have:

$$\sum_{v \in V(D)} \overline{\mathtt{lt}}(v) + \sum_{F \in F(D)} \mathtt{lt}(F) = 2|A(D)|.$$

Note that $\overline{\mathtt{lt}}(v) = 0$ if and only if v is a terminal (i.e. $v \in \mathcal{T}(D)$). Otherwise, $\overline{\mathtt{lt}}(v) \geq 2$.

$$\begin{aligned}
\sum_{F \in F(D)} (\mathtt{lt}(F) - 2) &\leq \sum_{F \in F(D)} (\mathtt{lt}(F) - 2) + \sum_{v \in V(D)} \max(0, (\overline{\mathtt{lt}}(v) - 2)) \\
&\leq \left(\sum_{F \in F(D)} \mathtt{lt}(F) \right) - 2|F(D)| + \left(\sum_{v \in V(D)} \overline{\mathtt{lt}}(v) \right) - 2|V(D)| + 2|\mathcal{T}(D)| \\
&\leq 2(|A(D)| - |F(D)| - |V(D)|) + 2|\mathcal{T}(D)| \\
&\leq 2|\mathcal{T}(D)| - 4
\end{aligned}$$

The last inequality follows from Euler's formula. □

From Theorem 7.3.5, it is easy to derive that $|AF(D)| \leq \tau(D) - 2$, which leads to the following.

Corollary 7.3.6. *For every plane DAG D, we have*

$$\sum_{F \in AF(D)} \mathtt{lt}(F) \leq 4\tau(D) - 8$$

The following shows how to enumerate completions of a face with a bounded number of local terminals.

Lemma 7.3.7. *Given a face f with $\mathtt{lt}(F)$ local terminals, there are $2^{O(\mathtt{lt}(F))}$ subsets of arcs linking local terminals that can be embedded inside F without crossings. Furthermore, those sets can be generated within $2^{O(\mathtt{lt}(F))}$ time.*

Proof. It is well known that the number of triangulations of a polygon with n angles corresponds to the Catalan number, and that it is $O(4^n)$ asymptotically [Wal72]. As any such triangulation has $2n - 3$ edges, there are $2^{O(n)}$ orientations for each of them, and there are $2^{O(n)}$ subgraphs for each of them. As $O(4^n) \times 2^{O(n)} \times 2^{O(n)} = 2^{O(n)}$, we hence have $2^{O(\mathtt{lt}(F))}$ sets of arcs to consider. Finally, note that all this can be done within $2^{O(\mathtt{lt}(F))}$ time. □

Combining Corollary 7.3.6 and Lemma 7.3.7 allows us to perform our branching in FPT time.

Corollary 7.3.8. *Given a DAG D, there are $2^{O(\tau(D))}$ structured completions of D within alternating faces, and those can be generated within $2^{O(\tau(D))}$ time.*

7.3.4 Computing the remaining solution in simple faces

At this point, we may assume from Subsection 7.3.3 that we are in a branch corresponding to an "optimal" structured completion across alternating faces. That is, if (D, k) is positive, for a structured solution X, we have guessed the restriction $X_{\mathcal{AF}}$ to the set of alternating faces, letting $k_{\mathcal{AF}} = |X_{\mathcal{AF}}|$. We will then consider the instance $(D + X_{\mathcal{AF}}, k - k_{\mathcal{AF}})$, which, for positive instances, admits a (structured) solution with all arcs embedded in simple faces (of D). Clearly, if (D, k) was negative, no guess of $X_{\mathcal{AF}}$ in alternating faces can yield a $(D + X_{\mathcal{AF}}, k - k_{\mathcal{AF}})$ having a solution (in particular within simple faces). Therefore, deciding if $(D + X_{\mathcal{AF}}, k - k_{\mathcal{AF}})$ admits a solution with all arcs belonging to simple faces of D is equivalent to instance (D, k) for k-PSCA.

We begin by recalling the existence of a polynomial-time algorithm computing minimum dijoins, then reduce the computation of a (remaining) minimum solution of $(D + X_{\mathcal{AF}}, k - k_{\mathcal{AF}})$ across simple faces to this problem over an auxiliary graph.

Computing a minimum dijoin

Given a digraph D, recall a *dijoin* of D is a subset of arcs the contraction of which renders D strongly connected. The following fundamental result of Frank [Fra81] comes from his algorithmic proof of the Lucchesi-Younger theorem (see 6.1.20).

Theorem 6.1.21 (Frank '81)**.** *There exists a polynomial-time algorithm computing a minimum dijoin of a digraph.*

We recall that a dijoin can equivalently be defined as a subset of arcs $X \subseteq A$ such that, letting $\overleftarrow{X} = \{(y, x) : (x, y) \in X\}$, $D + \overleftarrow{X}$ is strongly connected. Restating Theorem 6.1.21 yields a polynomial-time algorithm for computing a minimum $X \subseteq A(D)$ such that $D + \overleftarrow{X}$ is strongly connected. We now turn to showing that, after our guess in alternating faces, the completion in simple faces reduces to the above.

Reducing to the minimum dijoin problem

The following observation allows us to only consider a single possible completion for each simple face.

Observation 7.3.9. *For any simple face F, with local source s and local sink t, $\{(t, s)\}$ is the unique non-empty structured completion of F.*

Observation 7.3.9 means that the only question left is to decide *which* simple faces to complete, rather than how.

We define an auxiliary digraph D', starting from the k-subdivision of $D + X_{\mathcal{AF}}$. Then for every subdivided (initial) simple face F' of D', with local terminals s and t, we add the arc (s, t) to D', and call these arcs A'. The following lemma shows our reduction: the reversal of a minimum dijoin in D' is always a minimum size structured solution of $D + X_{\mathcal{AF}}$ within (initial) simple faces (with arcs "(t, s)").

Lemma 7.3.10. *For every subset $\overleftarrow{X'} \subseteq \overleftarrow{A'}$ with at most k arcs, $D + X_{\mathcal{AF}} + \overleftarrow{X'}$ is strong if and only if $D' + \overleftarrow{X'}$ is strong. Furthermore, any minimum subset $Y' \subseteq A(D')$ with at most k arcs and such that $D' + \overleftarrow{Y'}$ is strong, is such that $Y' \subseteq A'$.*

Proof. The first statement follows from the following facts, holding for any $X' \subseteq A'$,

- $D' + \overleftarrow{X'}$ can be obtained from $D + X_{\mathcal{AF}} + \overleftarrow{X'}$ by subdividing arcs,
- $D + X_{\mathcal{AF}} + \overleftarrow{X'}$ can be obtained from $D' + \overleftarrow{X'}$ by contracting arcs, and
- both operations preserve strong connectivity.

Let us now prove the second statement, and assume towards a contradiction that Y' contains an arc stemming from a k-subdivision, say of (u, v), in $D + X_{\mathcal{AF}}$. Let $(u = v_0, v_1, \ldots, v_k, v_{k+1} = v)$ be the dipath produced by the subdivision of (u, v), and let (v_i, v_{i+1}) be an arc of Y'. By the minimality of Y', there is a pair x, y of vertices requiring (v_{i+1}, v_i) to be connected by a dipath. We can assume, without loss of generality, that x and $y \neq v_j$, with $1 \leq j \leq k$. Indeed, if $x = v_j$ (resp. $y = v_j$) we can replace x by v (resp. y by u) as there exist a $v_j v$-dipath (resp. a uv_j-dipath) avoiding (v_{i+1}, v_i). Now, note that this xy-dipath should actually go through (v_{k+1}, v_k), (v_k, v_{k-1}), and so on until $(v_1, v_0 = u)$, but this would contradict that Y' has size at most k. □

Finally Lemma 7.3.4, Corollary 7.3.8, and Theorem 6.1.21, allow us to derive Theorem 7.1.3.

7.3.5 Adapting the strategy of k-Directed-PSCA

Non-acyclicty. The first hurdle one runs into when trying to generalize the approach used for k-Directed-PSCA is that instances of k-PSCA are not equivalent under condensation. Indeed, contracting arcs may increase the solution size, or even prevent augmentability, see Figure 7.3.

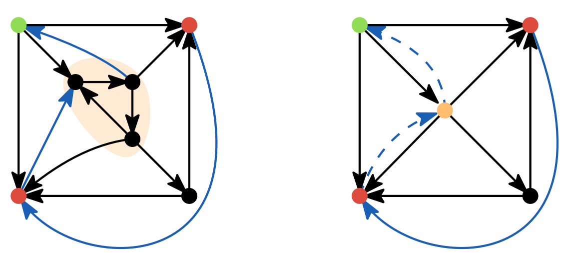


Figure 7.3: An instance of PSCA, with a non-trivial strong component in orange, and the oriented graph obtained by contracting it into a single vertex. The initial instance has a solution of size three, shown in blue, that is not preserved for the condensation, which is a negative instance.

As a result, the local terminals we define for k-PSCA will have size possibly linear in n (see Subsection 7.4.1), loosening the nature of structured solutions. While we define analogues of simple and alternating faces, the solutions may not be reconfigured as seamlessly anymore. In particular, simple faces may now admit several incomparable completions, which we will need to guess with a randomized algorithm.

Structured solutions avoiding digons. In k-PSCA, the main difficulty lies in defining a notion of structured completions of a face that avoids digons, but is still constrained enough to allow for bounds in terms of k. The obvious obstacle is that an arc of some solution X cannot be reconfigured onto a pair of vertices that already form an arc in the face at hand. More challenging is the fact that, when D is not 3-connected, there may exist arcs between vertices of F that are embedded outside of F, but still obstruct the "shifting" of arcs in X during reconfiguration. While such shifts always end up in local terminals in k-Directed-PSCA, this is not guaranteed to happen for instances of k-PSCA. Our goal is still to mimic this intuition, by considering "leftmost" or "rightmost" shifts of X toward local terminals of F, while avoiding digons. We achieve to define such structured solutions, which allows us to consider only a bounded number of structured completions in each face F.

Branching in alternating faces. Although the size of each local terminal can now be very large, their number across alternating faces can still be bounded by a function of k using same arguments as for k-Directed-PSCA (see Subsection 7.4.1). Then, thanks to our definition of structured completions, we may branch over a bounded number of possible completions across all alternating faces.

Guessing in simple faces. Having branched on a possible (structured) completions within alternating faces, we are left with the task of finding a minimum structured solution across all simple faces. In k-Directed-PSCA, this question was reducible to the Minimum Dijoin problem thanks to each simple face admitting an unique structured completion: the arc from its local sink to its local source, which rendered $D[V(F)]$ strong. This is not achievable in k-PSCA, where existing arcs from the local source to the local sink in D may prevent the addition of an arc in the other direction. In fact, a structured completion of a simple face may need multiple arcs, and may not even render $D[V(F)]$ strong by itself. In particular, there can be multiple structured completions of F that are pairwise incomparable in terms of connectivity. The crucial step is to show that this number can still be bounded by an absolute constant (see Subsection 7.4.6), but this prevents a direct reduction to Minimum Dijoin, or even Weighted SCA. Indeed, we would need to choose one completion for each face in advance, to prevent the algorithm from producing solutions with crossings. To overcome this, we guess a single allowed completion in each simple face uniformly at random, then show a reduction to the Weighted-SCA problem. When the "correct" restriction of a structured solution has been guessed across the *at most* k simple faces containing an arc of the solution, our reduction is guaranteed to find some minimum solution. Since this probability is only a function of k, our algorithm can be derandomized in FPT-time using universal sets.

7.4 FPT algorithm for k-PSCA

We begin in Subsection 7.4.1 by describing our instances, in particular defining local terminals, and the subwalks between them (interval dipaths). We show a few properties satisfied by minimal solutions in Subsection 7.4.2. Then, in Subsection 7.4.3, we show that certain "stacked" endpoints of a completion can be enumerated efficiently. Using this, we define the notion of structured solutions in Subsection 7.4.4, which are "maximally stacked" towards local terminals while maintaining strong connectivity. We show how to branch over all possible structured completions of alternating faces in Subsection 7.4.5, reducing k-PSCA to multiple instances of the same problem asking for completions within simple faces only. Then, Subsection 7.4.6 shows how to compute a remaining minimum solution in simple faces by a reduction to Weighted SCA, yielding our FPT algorithm for k-PSCA in Subsection 7.4.7.

7.4.1 Structure of the instance

The first structural property to note is that the arguments of Observation 7.3.1 also apply to k-PSCA, allowing us to bound the number of terminal components in terms of the solution size.

Observation 7.4.1. *For any positive instance (D, k), we have $|\mathcal{T}(D)| \leq 2k$*

For a face F, the subwalk of its boundary going clockwise from a (labelled) vertex a to a (labelled) vertex b is denoted by $[a, b]$. A *strong interval* I of F is a maximal subwalk of its boundary whose vertices belong to the same strong component of D.

Local terminals and face types

We may now define local terminals for k-PSCA, see Figure 7.4 for an illustration.

Definition 7.4.2 (local terminals)**.** Given a plane digraph D (possibly containing loops), and a face F of D, a strong interval $I = [a, b]$ is a *local terminal* if one of the following holds.

- The arc preceding a and the one following b along F are out-going from a and b respectively. In this case, I is a *local source.*
- The arc preceding a and the one following b along F are incoming towards a and b respectively. In this case, I is a *local sink.*

Let $\texttt{lt}(F)$ denote the number of local terminals around a face F. Note that non-terminal strong intervals have one incoming and one outgoing arc along the face, meaning local terminals alternate around the face, and are in even number. A face F is *strong* when $\texttt{lt}(F) = 0$, meaning it contains a single strong interval I containing all vertices of the face. Note in particular that we can omit any arcs of a solution embedded in it. Otherwise, a face F is *simple* when $\texttt{lt}(F) = 2$, and *alternating* whenever $\texttt{lt}(F) \geq 4$.

Interval dipaths

For a face F, an *interval dipath* is a maximal subwalk of the boundary, after removing local terminals, see Figure 7.4. We stress the arcs of an interval dipath are not necessarily oriented in the same direction. By definition, an interval dipath P is consecutive to a source component S and a sink component T. As a convention we will denote the vertices of P by $v_1, \ldots, v_r$, with v_1 and v_r adjacent to S and T, respectively. When referring to interval dipaths, as well as in the pictures, *left* and *right* will refer to vertices closer to S (i.e. with lower index), or closer to T (i.e. with higher index), respectively.

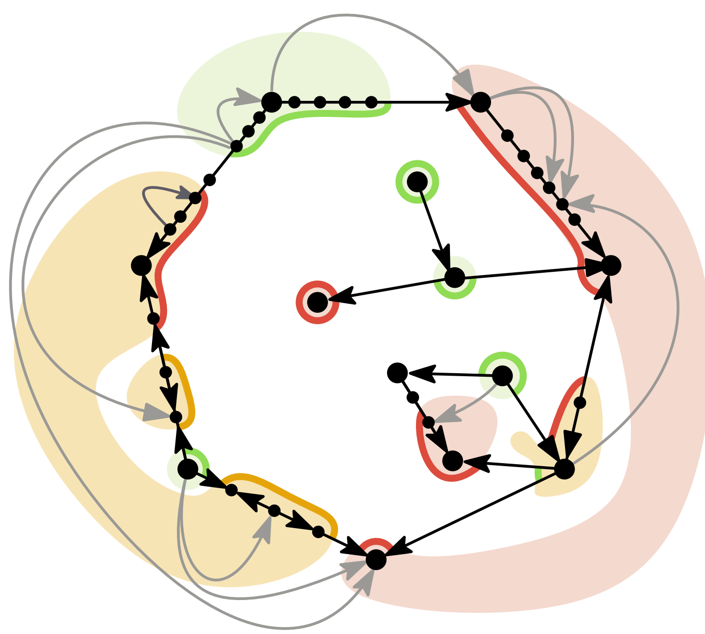


Figure 7.4: A face F of D, with arcs of $V(F)^2$ embedded outside F shown in gray. The strong components of D are shown as clouds in green (source), red (sink) and orange (non-trivial intermediate). Each consecutive intersection of these components with the boundary of F forms a strong interval. Strong intervals are shown by thick lines, in green for local sources, red for local sinks, and orange otherwise.

In order to define structured solutions in analogy to DIRECTED-PSCA, it would help for interval dipaths to be directed paths, in order to maintain a solution through our reconfiguration. This is not always the case, but they behave in the same way connectivity-wise, thanks to the following.

Lemma 7.4.3. *Consider any interval dipath $P = [v_1, v_r]$ of a face F, from local source S to (consecutive) local sink T.*

1. *For any vertex $s \in S$ (resp. $t \in T$) and any vertex $v_i \in P$, there exists a dipath from s to v_i (resp. from v_i to t).*
2. *For any $i, j \in [1, r]$ with $i < j$, there exists a directed path from w_i to w_j.*

Proof. Let v_0 be the vertex preceding v_1 around F. As s and v_0 belong to the same strong component, there is a dipath from s to v_0, and thus to v_1, as the arc between v_0 and v_1 is out-going v_0. To reach the other vertices of P we just apply the property 2, we prove now. The case of paths towards t is symmetrical.

For property 2, consider two vertices v_i and v_j, with $i < j$. Let $I_0, \ldots, I_q$ be consecutive strong intervals around F, such that $v_i \in I_0$ and $v_j \in I_q$ (with possibly $q = 0$). As these intervals are not local terminals (and given the orientation of (v_0, v_1)), the arcs along F between these strong intervals are oriented in the same direction, from I_j to I_{j+1}. As v_i and the rightmost vertex of I_0, v_x, belong to the same strong component, there is a dipath from v_i to v_x, and thus to v_{x+1}, and so on until reaching I_q, where the strong component containing I_q allows us to reach v_j. □

With respect to preserving a solution (that is preserving connections with dipaths), the above allows us to treat the interval dipaths between the local terminals as dipaths.

Bounding the number of local terminals in alternating faces

Let $AF(D)$ denote the set of alternating faces of D.

Lemma 7.4.4. *For every plane digraph D, we have*

$$\sum_{F \in AF(D)} \mathtt{lt}(F) \leq 4|\mathcal{T}(D)| - 8$$

Proof. Successively contracting any (non-loop) arc lying within a strong component of D, yields a plane digraph D' that is a DAG with loops. More precisely, deleting the loops of this digraph yields an oriented DAG. Recall that the definitions of local terminals and alternating faces also hold for DAGs with loops.

Claim 7.4.1. Given a digraph D_1 and a digraph D_2, obtained from D_1 after contracting a single non-loop arc uv lying inside a strong component of D_1, we have

$$|\mathcal{T}(D_1)| = |\mathcal{T}(D_2)| \quad \text{and} \quad \sum_{F \in AF(D_1)} \mathtt{lt}(F) = \sum_{F \in AF(D_2)} \mathtt{lt}(F).$$

Proof of the claim. Note that the strong components of D_1 are bijectively mapped to the strong components of D_2, in such a way that the terminal components of D_1 are mapped to the terminal components of D_2. Also, the faces of D_1 are bijectively mapped to the faces of D_2. These bijections are such that if a face F_1 of D_1 is mapped to a face F_2 of D_2, we have $\texttt{lt}(F_1) = \texttt{lt}(F_2)$. This implies that the faces of $AF(D_1)$ are mapped to those of $AF(D_2)$. Hence, the claim holds. $\diamond$

It is thus sufficient to prove the lemma for the plane oriented DAGs with loops. Let us proceed by induction on the number of these loops. The initial case, with a loopless D, follows from Corollary 7.3.6, and we distinguish two cases for the induction. First, consider the case where there exists a face F with only one arc on its boundary, that is, a loop $e = (u, u)$. Then delete e, and observe that for the other face incident to e, the number of local terminals is unchanged, such as for all the other faces. Hence, by induction the inequality of the lemma holds.

Otherwise, we may consider the case where a loop $e = (u, u)$ separates two non-empty regions. Let D_1 and D_2, be the subdigraphs of D induced by the arcs inside or outside ℓ, respectively. One can observe that by definition, ℓ is contained in a strong interval for each of its incident faces. Hence, for any face F of D, the corresponding face of D_1 or D_2, according to the position of F with respect to ℓ, keeps the same number of local terminals. Hence,

$$\sum_{F\in AF(D)} \texttt{lt}(F) = \sum_{F\in AF(D_1)} \texttt{lt}(F) + \sum_{F\in AF(D_2)} \texttt{lt}(F).$$

Furthermore, if the strong component C of v in D (which has only v as vertex) is terminal in D, then in each of D_1 and D_2 the strong components of v, C_1 and C_2 respectively, are also terminal in their respective digraphs, while if C is not terminal at least one of C_1 or C_2 is not terminal. Hence, $|\mathcal{T}(D_1)| + |\mathcal{T}(D_2)| \leq |\mathcal{T}(D)| + 1$, which implies that:

$$\sum_{F\in AF(D)} \texttt{lt}(F) \leq 4|\mathcal{T}(D_1)| - 8 + 4|\mathcal{T}(D_2)| - 8 \leq 4|\mathcal{T}(D)| - 8.$$

$\square$

7.4.2 Properties of minimal solutions

To come up with a definition of structured solutions, enabling an FPT enumeration, we must be able to constrain our completions. After deriving some properties imposed by minimality,

The main obstacle we wish to avoid in reconfiguration is the creation of a digon involving X and an arc of D. Therefore, from now on, we consider *completions* as *multi*completions, that is X can induce some $D + X$ that contains multi-arcs, digons, and loops, as long as those are fully within X. The following lemma shows that this relaxation is sound: when X is a minimal (multi)solution, it cannot contain any multi-arc, digon or loop, meaning it is indeed a (proper) solution.

Lemma 7.4.5. *For any digraph D, and any minimal (multi)completion X such that $D + X$ is strongly connected, X contains at most one arc between any pair of strong components of D. In particular, X cannot contain any loop, digon, or multi-arc.*

Proof. Towards a contradiction, assume that there are two strong components in D, C_1 and C_2, linked by two arcs of X, e_u and e_v. Assume without loss of generality that $e_u = (u_1, u_2)$ goes from C_1 to C_2. First, note that e_v does not go from C_1 to C_2, as otherwise e_u could be replaced by a dipath in C_1 from u_1 to e_v, followed by a dipath in C_2 from e_v to u_2. We hence assume now that $e_v = (v_2, v_1)$ goes from C_2 to C_1.

Consider now the digraph $D' = D + (X - \{e_u, e_v\})$, which is not strong, as X is minimal. Since $D + X$ is strong, D' has exactly two terminal components C'_1 and C'_2, with $C_1 \subseteq C'_1$ and $C_2 \subseteq C'_2$. As D, and thus D', is connected, there is a dipath from the unique source component of D', say C'_1, to the unique sink component of D', C'_2. In that case, adding the arc e_v would suffice to turn D' strong, a contradiction. □

Our reconfiguration mainly operates on subwalks W in which elements of X only have one endpoint. While Lemma 7.4.5 already implies this when W is a local terminal, this is not necessarily the case for interval dipaths. Given an interval dipath P, the arcs of X_P having both endpoints on P are called *backward arcs*, but these can still be constrained as follows.

Observation 7.4.6. *If X is a minimal solution, for any backward arc $e = (w_j, w_i)$ of an interval dipath P:*

(i) no arcs of X have both endpoints in $[w_i, w_j]$, that is, backward arcs are not nested;

(ii) $i < j$, that is, e is directed from right to left.

7.4.3 Endpoints of leftmost and rightmost arcs on a subwalk

In the following, we show that endpoints of a solution X that are "stacked" on the ends of a given subwalk W, in practice a local terminal or an interval dipath, can be guessed efficiently.

Definition 7.4.7. Let $W = (w_1, \ldots, w_r)$ be a subwalk of F, such that every arc of X_F has at most one endpoint on W. Then, the arcs of X_W are said to be in *leftmost position* (resp. *rightmost position*) if shifting any of their W-endpoints further left (resp. right) on W does not yield a completion. That is, if replacing any (underlying) arc $e = \{w_i, x\}$ of X_W with an $e' = \{w_h, x\}$ such that $h < i$ (resp. $h > i$) results in a digraph that is either not planar, or not oriented.

The main purpose of Definition 7.4.7 is that (a bounded number of) endpoints that are in leftmost or rightmost position on a given W can only lie on a bounded number of vertices, which can be enumerated.

Lemma 7.4.8. *Given a subwalk W around a face F of an oriented graph D, for any $q \geq 0$, there exists a set $Z^L_W(q)$ (resp. $Z^R_W(q)$) of q-tuples $(w_{i_1}, \ldots, w_{i_q})$ of vertices of P such that:*

1. *$Z^L_W(q)$ (resp. $Z^R_W(q)$) contains at most 5^q tuples, and can be enumerated, given D, W, and q in $(5^q n^{O(1)})$ time*

2. *For any set X_W of arcs in leftmost position (resp. rightmost position) along W, with $q = |X_W|$, there exists a tuple $(w_{i_1}, \ldots, w_{i_q}) \in Z^L_W(q)$ (resp. $\in Z^R_W(q)$), such that the j^{th} arc of X^L_W, from left to right (resp. from right to left) has an endpoint on w_{i_j}.*

Proof. By symmetry, we show this for the left side only, and we proceed by induction on q. If $q = 1$, the first two elements of $Z^L_W(1)$ are w_1, and w_2. Now, if there exists an angle u around F from which one cannot attach an arc with w_1, nor w_2, it is either because u corresponds to the same vertex as w_1 or w_2, or because u is a common neighbour of w_1, and w_2, but there is at most one such vertex around F, as $D[V(F)]$ is an outerplanar graph with $\{w_1, w_2\}$ on its boundary. For each of these three vertices, we add in $Z^L_W(1)$ their leftmost non-neighbour on P.

Now, assume we have defined such a set $Z_W^L(q)$ for a given q. We are going to construct $Z_W^L(q+1)$, by considering every q-tuple of $Z_W^L(q)$ and extending it, in at most 5 different ways. We hence consider a particular tuple, and we denote w_i its last vertex. We first prolong the tuple in two different ways, in the first one we add w_i (the tuples allow repetitions), and in the second one we add w_{i+1}. Now, if there exists an angle u around F from which one cannot attach an arc with w_i, nor w_{i+1}, it is either because u corresponds to the same vertex as w_i or w_{i+1}, or because u is the common neighbour of w_i, and w_{i+1}. For each of these three vertices, we prolong the tuple with their leftmost non-neighbour on P, after w_i. □

7.4.4 Structured solutions

In this subsection, we show that every positive instance (D, k) admits a structured solution, defined in Definition 7.4.16. What we wish out of structured solutions is, for every local terminal or interval dipath, to "maximize" P-leftmost and P-rightmost arcs, the endpoints of which can be enumerated through Lemma 7.4.8.

The first observation is that constraining solutions as such is easy for local terminals. Indeed, for every subwalk W, corresponding to a local terminal, $V(W)$ belongs to the same strong component of D, meaning any substitution of the endpoints of X_W in W maintaing strong connectivity. Therefore, we can ask for those arcs to be in leftmost (or rightmost) position.

Observation 7.4.9. *For every solution X, and every local terminal W, the endpoints of X_W (and only those) can be reconfigured to be in leftmost position while maintaining that X is a solution.*

Applying Observation 7.4.9 (independently) to each local terminal means we can already ask for solutions that are in leftmost position in each, and thus guessable. This is indeed how we define "structured" for endpoints on local terminals in Definition 7.4.16, but before being able to define structured solutions fully, we turn to the case of interval dipaths. For an interval dipath P, arcs of X_P cannot always be shifted to all be in leftmost or rightmost position on P. Still, we can split P as follows to recover "simultaneously" leftmost and rightmost arcs on P

Definition 7.4.10. Given a solution X, and an interval dipath P, the *left subwalk* P^L with respects to X is the longest containing v_1 such that:

1. every arc of X_P has at most one endpoint in P^L (i.e. P^L has no backward arc), and
2. the arcs of $X_P^L := X_{P^L}$ are in leftmost position, then, these are called the *P-leftmost arcs* of X.

Starting with $P - V(P^L)$, we symmetrically define the *right subwalk* P^R and the subset X_P^R of *P-rightmost arcs.*

Now, by Lemma 7.4.8, for any solution X, the endpoints of X on P^L and P^R can be guessed in FPT time. Solutions may not always be reconfigurable such that all endpoints lie on P^L or P^R for each interval dipath P. We therefore also need to consider "middle" arcs.

Definition 7.4.11. Given a solution X, and an interval dipath P, the *middle subwalk* of P with respects to X is defined as $P^M := P - (V(P^L) \cup V(P^R))$. Then, the set of *$P$-middle arcs* X_P^M is the subset of X_P having at least one endpoint in P^M.

Reconfiguring middle arcs on interval dipaths

Our goal is to minimize X_P^M, then, we will also need to show how to reconfigure its endpoints into ones that can be enumerated as well.

Definition 7.4.12. Formally, for any interval dipath P, a solution X *locally minimizes* X_P^M, if there is no minimum solution X' such that:

- $X \setminus X_P^M \subseteq X'$,
- $X' \setminus X$ is obtained from arcs of X_P^M by replacing its endpoints on P (and preserving those outside),
- $|X'^M_P| < |X_P^M|$.

In the following, we consider solutions that locally minimize X_P^M simultaneously for every interval dipath P. This is possible by reconfiguring X_P^M for any interval dipath which doesn't satisfy the second item Definition 7.4.12. The process indeed finishes, because the sum of $|X_P^M|$ strictly decreases at each step.

The first result we need is that endpoints of X_P^M on P can always be gathered onto a common (labelled) vertex while maintaining a completion. Still, this operation is defined without considering strong connectivity, and the resulting completion may not be a solution.

Lemma 7.4.13. *Consider a face F of D, a completion X, and two (labelled) vertices u and v around F. Assume X is incident at u with arcs X_u, and at v with arcs X_v, but is not incident to $]u,v[$. Then, replacing the u-end of the arcs of X_u with v, or replacing the v-end of the arcs in X_v with u, results in a completion X' such that any loops, digons, and multiple arcs of $D + X'$ are fully in X'.*

Proof. The following considers all arcs irrespective of their orientation, and we use xy to mean both (x,y) and (y,x). We let X_u and X_v refer to the endpoints of X incident to u and v respectively, throughout the proof. Consider successively replacing the u endpoint of arcs in X_u with v. If this process can be done to obtain the result, we are done. Otherwise, we stop the process at some xu where the above fails. This must be because the arc xv would produce a multi-arc or a digon with an already existing arc xv of D. Observe that at this point, any yv in (the updated) X_v is such that $y \in]v,x[$, else yv would cross xu in X. Therefore, no such y can be a neighbour of u in D, for otherwise yu would cross xv in D. Then, consider symmetrically the successive substitution of the v endpoints of X_v with u. Since u is not a neighbour of any non-v endpoint y of X_v, all substitutions apply, and all u and v endpoints of the initial $X_u \cup X_v$ are now u. Note that the only created loops, digons, and multi arcs, are fully in the resulting ☐

The second tool is a property on strong components of $D - e$ for $e \in X_P^M$, which will enable us to recover strong connectivity after an application of Lemma 7.4.13.

Lemma 7.4.14. *Given a positive instance (D,k), a face F of D and an interval dipath $P = (v_1,\ldots,v_r)$, consider a minimum solution X that locally minimizes $|X_P^M|$. If $X_P^M \neq \emptyset$ there is a vertex v_i in P^M incident to an arc $e \in X_P^M$. Let v_i and e be the leftmost (resp. rightmost) among such vertices and arcs. Then, we have the following properties:*

1. *The arc e is such that $e = (v_i, x)$ (resp. $e = (x, v_i)$) for some vertex x that can possibly belong to P^L (resp. to P^R).*

2. The local terminals contiguous to P, belong to distinct strong components in $D + X - \{e\}$.

Proof. For property 1, assume towards a contradiction that $e = (x, v_i)$. By maximality of X_P^L, it is possible to replace e in $D+X$ by some arc (x, v_h), with $h < i$, and while remaining a plane oriented graph. Let v_h, be the leftmost such vertex. Note that the new obtained digraph is also strong, as for any dipath going through $e = (x, v_i)$, we can replace this arc by a dipath going through (x, v_h) and through a dipath from v_h to v_i (by Lemma 7.4.3). This contradicts the maximality of $|X_P^L|$.

For property 2, we can assume, towards a contradiction, that the local source S and the local sink T contiguous to P belong the same strong component C' of $D' = D + X - \{e\}$. Note that by Lemma 7.4.3, $V(P)$ also is contained in C', including v_i. If e was a backward arc, this would contradict the minimality of X, as its two endpoints belong to C'. Hence assume now that $x \notin C'$. This implies that C' is a sink component. As $e \notin X_P^L$, it is possible to replace $e = (v_i, x)$ with an arc $e' = (v_h, x)$, with $h < i$ while remaining an oriented plane graph. Let h be the minimum such value. Let us now observe that this results in a strong oriented graph, as any dipath going through $e = (v_i, x)$, can now go through a dipath from v_i to v_h contained in C', and then through $e' = (v_h, x)$. Hence, this results in a solution $X - \{e\} + \{e'\}$ that is still minimum, but that has a larger set of arcs in leftmost position, a contradiction. □

We are now ready to combine Lemma 7.4.13 and Lemma 7.4.14 to impose "guessable" endpoints on each interval dipath.

Lemma 7.4.15. *Every positive instance (D, k) admits a solution X where, for any interval dipath $P = (v_1, \ldots, v_r)$, there is a tuple $(v_{i_1}, v_{i_2}, \ldots) \in Z_P^L(|X_P^L| + |X_P^M|)$ such that:*

- *the endpoints of the arcs in X_P^L are $v_{i_1}, \ldots, v_{i_{|X_P^L|}}$ successively,*
- *Arcs in X_P^M have a common endpoint, among $v_{i_{|X_P^L|+1}}, \ldots, v_{i_{|X_P^L|+|X_P^M|}}$.*

In particular there is no backward arc with both ends in P^M.

Proof. Consider a minimum solution X that locally minimizes $|X_P^M|$, for every interval dipath P of D. Now for a particular P around a face F, consider the arcs of X_P^M, if there are some, and let $e_1 = (v_i, x_1)$ and $e_w = (x_w, v_j)$ be its leftmost and rightmost arcs, respectively (by Lemma 7.4.14.1).

Claim 7.4.2. The digraph $D + X - X_P^M$ contains a dipath from x_1 to S, the local source contiguous to P in D, and a dipath from T, the local sink contiguous to P in D, to x_w.

Proof of the claim. By the minimality of X, the digraph $D + X - e_1$ is not strong and as e_1 suffices to make it strong, $D + X - e_1$ has exactly one source component containing x_1, and one sink component containing v_i. We denote them S_{x_1}, and T_{v_i}. Note that by Lemma 7.4.3, T_{v_i} contains $[v_i, v_r]$ and T. Now, since S_{x_1} is the unique source component, there exists a dipath from x_1 to any vertex, and in particular there is a dipath P' from x_1 to S. If this dipath P' were to use any vertex $v_j \in P^M$. As $v_j \in T_{v_i}$, the path P' would contain a subdipath from T_{v_i} to S, implying that T_{v_i} also contains S. This contradicts Lemma 7.4.14.2. Hence, P' does not go through a vertex of P^M, which in turn implies it does not go through an arc of X_P^M. The symmetrical proof yields the dipath from T to x_w. ◇

Let us denote $e_1, \ldots, e_w$ the arcs of X_P^M with exactly one endpoint on P^M (we will see that arcs with both ends on P^M are not needed in the solution), from left to right, with $e_j = \{v_{i_j}, x_j\}$

for some vertex $v_{i_j} \in P^M$ and some vertex $x_j \notin P^M$. By Lemma 7.4.8 it is possible to push their endpoints along P further left on vertices of some tuple of $Z_P^L(|X_P^L|+|X_P^M|)$ (the first vertices of the tuples corresponding to the endpoints of X_P^L), while preserving simplicity and planarity, but maybe losing strong connectivity. Then, we apply Lemma 7.4.13 to gather these endpoints on a common vertex v_h that was an endpoint of the tuple. Let ${X'}_P^M$ be the arcs reconfigured from X_P^M as such, and note that ${X'}_P^M$ is now a plane completion that may contain digons, multiple arcs and loops, but those are fully in X. The following claim ensures that this gathering restores strong connectivity, and in particular thanks to Lemma 7.4.5, ensures the resulting completion X' is indeed a (proper) solution.

Claim 7.4.3. For every such vertex v_h, the oriented plane graph $D + X - X_P^M + {X'}_P^M$ is strong.

Proof of the claim. Let $C_1, ..., C_w$ denote the strong components of $x_1, \ldots, x_w$ in $D + X - X_P^M$. Note that some of these components may coincide. Actually, this list of strong components contains all the terminal components of $D + X - X_P^M$. Indeed, any other source component should contain an endpoint of X_P^M, that is a vertex of P. This would imply that this source would contain S and thus x_1 (by Claim 7.4.2), so it is C_1. The same applies to sink components and x_w. Now, the reconfiguration of all endpoints onto a common vertex v_h means that any (previous) source component C_i now has an arc from v_h, and any (previous) sink C_j has an arc towards v_h, that is, all sink components now reach all source components, which implies strong connectivity. ◇

Hence, $X - X_P^M + {X'}_P^M$ is the desired solution. □

Existence of structured solutions

We can now define structured completions, and thus structured solutions. This definition is based on Lemma 7.4.8, following the constraints we showed can be imposed on endpoints on each local terminal and interval dipath.

Definition 7.4.16. A completion X of D is *structured* if for every face F of D we have the following:

- For every local terminal S (resp. T), the endpoints of X_S (resp. X_T) are in leftmost position (clockwise). Hence their endpoints correspond to some tuple of $Z_S^L(|X_S|)$ (resp. $Z_T^L(|X_T|)$).
- For every interval dipath P, the endpoints of X_P^R are prescribed by some tuple of $Z_P^R(|X_P^R|)$.
- For every interval dipath P, the endpoints of $X_P^L \cup X_P^M$ are prescribed by some tuple of $Z_P^L(|X_P^L|+|X_P^M|)$. The endpoints of X_P^L correspond to the first values of this tuple, while the common endpoint of the arcs of X_P^M is one of the remaining vertices of the tuple.

Now, combining Observation 7.4.9, Definition 7.4.10 and Lemma 7.4.15 allows us to conclude the existence of structured solutions.

Theorem 7.4.17. *Every positive instance (D, k) admits a structured solution.*

7.4.5 Branching over structured completions of alternating faces

In the following, we show that the number of “patterns” of completions restricted to alternating faces can be bounded in terms of k, allowing us to enumerate them in FPT time.

Given an instance (D, k) and a structured solution X the *pattern* of X, denoted $\mathbb{P}(X)$, is the union of *facial patterns* $\mathbb{P}(X, F)$, over each face F of D. For a given face F, $\mathbb{P}(X, F)$ is the digraph obtained from the anti-directed cycle with $\mathtt{lt}(F)$ vertices, by subdividing each arc thrice, and by adding arcs inside, in order to sketch the position of the arcs of X_F inside F. Before adding these arcs we have to name the vertices of $\mathbb{P}(X, F)$. Denoting the succession of local terminals and interval dipaths around F clockwise with $S_1, P_1, T_2, P_2, S_3, \ldots, T_{\mathtt{lt}(F)}, P_{\mathtt{lt}(F)}$, the outerboundary of $\mathbb{P}(X, F)$ is the $(4\mathtt{lt}(F))$-cycle $(s_1, p_1^L, p_1^M, p_1^R, t_2, p_2^R, p_2^M, p_2^L, s_3, \ldots, t_{\mathtt{lt}(F)}, p_{\mathtt{lt}(F)}^R, p_{\mathtt{lt}(F)}^M, p_{\mathtt{lt}(F)}^L)$. Now, an arc $e \in X_F$ with its head (resp. its tail) in S_i, P_i^L, P_i^M, P_i^R, or T_i, is represented by an arc with its head (resp. its tail) on s_i, p_i^L, p_i^M, p_i^R, or t_i.

The purpose of the following lemma is to enumerate the patterns of a solution X inside a face F, taking $\mathtt{lt} = \mathtt{lt}(F)$ and $k_F = |X_F|$ in the following.

Lemma 7.4.18. *Given values $\mathtt{lt}, k_F$ there are $2^{O(\mathtt{lt}+k_F)}$ outerplanar digraphs allowing multiple arcs, whose outerboundary is a $4\mathtt{lt}$-cycle, and whose number of arcs is $4\mathtt{lt} + k_F$. Furthermore, those can be generated in $\left(2^{O(\mathtt{lt}+k_F)}(\mathtt{lt} + k_F)^{O(1)}\right)$ time.*

Proof. There are $O(4^{4\mathtt{lt}})$ undirected and simple outerplanar triangulations on $4\mathtt{lt}$ vertices [Wal72]. These triangulations have $8\mathtt{lt} - 3$ edges, so there are $\binom{k_F+8\mathtt{lt}-3}{k_F} = 2^{O(\mathtt{lt}+k_F)}$ ways of attributing to each edge a weight reflecting the multiplicity of arcs. Finally there are 2^{k_F} ways to orient the edges not on the outerboundary. So there are indeed, $2^{O(\mathtt{lt}+k_F)}$ such outerplanar digraphs. Finally it is clear that those can be enumerated in $\left(2^{O(\mathtt{lt}+k_F)}(\mathtt{lt} + k_F)^{O(1)}\right)$ time. □

Note that we could have restricted more the family of outerplanar graphs considered, by constraining the multiplicity of arcs, but this would not change their number asymptotically.

Now that we can guess the pattern of a structured solution (if the instance is positive), we have to show how to recover an actual solution.

Lemma 7.4.19. *There is an algorithm running in time $O(2^{O(k_F)}n^{O(1)})$, taking F and $\mathbb{P}(X, F)$, and returning a set $\mathcal{X}_F$ of $2^{O(k_F)}$ completions of F containing all restrictions of size k_F of structured solutions to F.*

Proof. Let X be any structured solution. According to Lemma 7.4.8, for each subwalk W corresponding to a local terminal or an interval dipath, the (say) q endpoints of X on W correspond to a q-tuple in list of $2^{O(q)}$ tuples. Hence, combining all these possibilities leads to $2^{O(k_F)}$ arc sets. The complexity of this algorithm clearly follows from the complexity of constructing those sets of tuples (see Lemma 7.4.8). □

Knowing the (full) pattern $\mathbb{P}(X)$ of a structured solution, this would lead to a $O(2^{O(k)}n^{O(1)})$-time algorithm to recover X. Unfortunately, this does not suffice as we are not able to guess the (full) pattern $\mathbb{P}(X)$ within FPT time. Still, the following shows we can guess the restriction of a pattern to the set of alternating faces. In turn, this allows us to branch over all their structured completions, which reduces k-PSCA to the computation of a solution within simple faces.

Theorem 7.4.20. *There is a $2^{O(k)}n^{O(1)}$-time algorithm, taking an instance (D, k) for k-PSCA, that either correctly outputs* no*, or returns a list $\mathcal{I} = \{(D_i, k_i)\}_i$ of instances such that:*

- $|\mathcal{I}| = 2^{O(k)}$,
- *D_i is a supergraph of D with $V(D_i) = V(D)$,*

- $k_i \leq k$, *and*
- (D, k) *is a positive instance of k-PSCA if and only if one of the instances* $(D_i, k_i) \in \mathcal{I}$ *admits a solution* X_i *whose arcs are all embedded in simple faces.*

Proof. We begin by computing in polynomial time the terminal components $\mathcal{T}(D)$ of D, and derive the resulting local terminals and interval dipaths in each face. First, if $k < \frac{1}{2}|\mathcal{T}(D)|$, we may correctly output `no` by Observation 7.4.1. In the following, we may thus assume $|\mathcal{T}(D)| \leq 2k$, which implies there are at most $O(|\mathcal{T}(D)|) = O(k)$ alternating faces by Lemma 7.4.4. So, there are (only) $2^{O(k)}$ ways to assign them the number of arcs of the solution that are embedded inside each of them. For each of these assignments, then for each alternating face F with k_F arcs of the solution allowed, we generate all the possible patterns $\mathbb{P}(X, F)$, for a solution X. By Lemma 7.4.18, those are $2^{O(\mathtt{lt}(F)+k_F)}$. Then, each pattern corresponds to at most $2^{O(k_F)}$ possible structured completions of F, which can all be enumerated in time $2^{O(k_F)}n^{O(1)}$ by Lemma 7.4.19. Putting this all together, we obtain $2^{O(k)}$ structured completions $(X_i)_i$ of the alternating faces, and for every i we create an instance $(D_i, k_i) = (D + X_i, k - |X_i|)$. It is now clear that if a solution to (D, k) exists, it should extend some X_i with arcs embedded in simple faces of D, yielding the last item of the statement for D_i. Conversely, if any (D_i, k_i) admits a solution X'_i with arcs embedded in (initial or new) simple faces, then $X_i + X'_i$ is a solution to (D, k). □

7.4.6 Computing a solution in simple faces

Because the number of simple faces cannot be bounded by a function of k, we cannot branch further on their completion. Instead, we will rely on the following constant bound on the number of restrictions of a minimum structured solution.

Lemma 7.4.21. *For any minimum solution X of D, and any simple face F of D,* $|X_F| \leq 12$.

Proof. We know from Lemma 7.4.5 that X_F has at most one arc between S and T, and it is also clear that there is at most one arc between S or T, and P_1 or P_2. Indeed, if there were several arcs between S and P_1, say, it would be sufficient to keep the one that goes the furthest on P_1, and orienting it towards S if it is not. Indeed, this suffices to make this subgraph of F strong, and there is hence no need of more arcs inside this subgraph. We have seen that each of P_1, P_2 has at most two backward arcs (c.f. Lemma 7.4.15).

It is hence sufficient to bound the number of arcs between P_1 and P_2. This is immediate if one of them is empty, so we assume that these paths contain at least one vertex. Let $P_1 = (u_1, \ldots)$ and $P_2 = (v_1, \ldots)$. We make the assumption that there is an arc from S to T as otherwise adding a single arc from T to S would make $D[V(F)]$ strong and there would be no need of more arcs to connect vertices inside F. This arc from S to T ensures that we can add every possible arc between P_1 and P_2, while remaining planar inside F. If there are too many arcs (i.e. more than three) between P_1 and P_2, let $e = \{u_i, v_j\}$ be the leftmost, and let $e' = \{u_{i'}, v_{j'}\}$ be the rightmost, and replace all these arcs by simply the following three arcs : $(u_{i'}, v_{j'})$, $(v_{j'}, u_i)$, and (u_i, v_j). Indeed, those three arcs turn $D[[u_i, u_{i'}] \cup [v_j, v_{j'}]]$ strong, so there would be no need of more arcs to connect vertices among those intervals. □

Note that actually, the 12 in Lemma 7.4.21 could be reduced to 3, at the price of a longer case by case analysis. In any case, having a constant number here is enough to reduce, by Lemma 7.4.18, the number of possible patterns $\mathbb{P}(X, F)$ for a structured solution X and a simple face F to a constant number. This in turn bounds the number of possible restrictions of minimal structured solutions, the set $\mathcal{X}_F$ returned by Lemma 7.4.19.

Corollary 7.4.22. *There exists an absolute constant N such that, for any instance (D, k), and any simple face F of D, the set $\mathcal{X}_F$ has size at most N.*

Let us now recall the Weighted SCA problem, to which we will reduce, and which admits a $2^{O(k \log k)} n^{O(1)}$ FPT algorithm, see Theorem 6.1.19.

(Weighted) Strong Connectivity Augmentation (SCA)
Parameter: k
Input: A digraph $D = (V, A)$, a set of links $L \subseteq V^2$, a weight function $w : L \to \mathbb{R}^+$, an integer k and a budget $\alpha \in \mathbb{R}^+$.
Question: Is there a subset $X \subseteq L$ such that $D + X$ is strongly connected, $|X| \leq k$ and $\sum_{e \in X} w(e) \leq \alpha$?

Theorem 7.4.23. *There is a $2^{O(k \log k)} n^{O(1)}$-time algorithm, taking an instance (D, k) for k-PSCA, and deciding if there exists a solution with at most k arcs all embedded in simple faces.*

To process the list $\mathcal{I}$ of Theorem 7.4.20, we could restrict to the simple faces that were originally (i.e. before applying the algorithm of Theorem 7.4.20) simple faces. As we have no control over the number of simple faces, the algorithm above would be computing "too many" solutions, but all of them are valid.

Proof. From (D, k) we will generate several instances (D, k, Y) where Y corresponds to allowed supplementary arcs. This list of instances will be such that (D, k) is positive if and only if one of the instances is. To solve these instances we just have to call the algorithm of Theorem 6.1.19, which has $2^{O(k \log k)} n^{O(1)}$ running time. The set Y being such that $D + Y$ is an oriented plane graph, a solution of (D, k, Y) will indeed be a solution of (D, k). Let us now see how to build this family of instances.

Consider the following randomized algorithm. For each simple face F pick arbitrarily at random, a set of $\mathcal{X}_F$ to add in Y. Then, note that $D + Y$ is a plane oriented graph, and that a structured solution X with arcs only in simple faces, if it exists, is contained in Y with probability at least $1/N^k$. This follows from the fact that $|X_F| > 0$ for at most k faces, and the fact that $X_F \subseteq Y_F$ with probability at least $1/|\mathcal{X}_F| \geq 1/N$. Thus constructing sets Y randomly, it would be enough in expectation for a positive instance (D, k) to make $2^{O(k)}$ tries Y to get a solution.

This approach can be derandomized by using universal sets. Those form a family H of functions h from the set of simple faces $F \in SF(D)$, to $\mathcal{X}(F)$, so that for any k faces $F_1, \ldots, F_k \in SF(D)$, and any k elements $Y_{F_i} \in \mathcal{X}(F_i)$, there is a function $h \in H$ such that $h(F_i) = Y_i$, for every $i \in [1, k]$. It is well known that there exist explicit construction of such family H of size $2^{O(k \log k)} \log f$ (see Theorem 2.5 in [UV22]), where f denotes the number of simple faces. By planarity we have $f = O(n)$, and so we have the desired complexity for our algorithm. □

7.4.7 FPT algorithm

Combining Theorem 7.4.20 and Theorem 7.4.23 yields the fixed-parameter tractability of k-PSCA.

Theorem 7.1.2. *k-Plane Strong Connectivity Augmentation is FPT with respect to k, and admits a $2^{O(k \log k)} n^{O(1)}$ algorithm.*

Proof. Given an instance (D, k) to k-PSCA, we use Theorem 7.4.20 to either output No, or to reduce the problem in time $2^{O(k)} n^{O(1)}$ to that of deciding whether one instance of a list $\mathcal{I}$ admits a solution within simple faces. $\mathcal{I}$ contains $2^{O(k)}$ instances, and for each $(D_i, k_i) \in \mathcal{I}$, we run the algorithm of Theorem 7.4.23 deciding such a solution in time $2^{O(k \log k)} n^{O(1)}$. Combining both steps yields a $2^{O(k \log k)} n^{O(1)}$ algorithm deciding (D, k). □

7.5 NP-completeness

In this section, we show that PSCA is NP-complete. The problem is clearly in NP as, given a plane digraph D and a set of (embedded) arcs X, one can verify in polynomial time that $D + X$ is strongly connected and plane. We show NP-hardness by a reduction from Planar 3-SAT with a variable cycle, which has been shown to be NP-complete by Lichtenstein [Lic82]. In this restriction, each clause contains exactly three literals, the incidence graph of the formula is planar, and a cycle passing through all variables may be added while preserving planarity. We note that our reduction can be adapted to yield acyclic instances, showing hardness even for plane DAGs.

Consider an instance $\Phi = \bigwedge_{j=1}^{m} \phi_j$ of Planar 3-SAT, on variables $x_1, ..., x_n$, which we take with a fixed embedding. We construct a plane oriented graph D such that Φ is satisfiable if and only if D is positive for PSCA (without budget). That is, there exists a set $X \subseteq V(D)^2$, such that $D + X$ is oriented, strongly connected and plane when preserving the embedding of D.

In the following, we build variable gadgets admitting exactly two maximal valid completions, corresponding to the negative and positive assignments. Clause gadgets consist of a single internal source component, incident to the three corresponding variable gadgets. All of their inner faces will be triangles, forbidding internal completions. The gist of the reduction is that in any augmentation of D, the source component of each clause gadget may only be attained by vertices outside of it if one of the corresponding literal gadgets is completed as to satisfy the clause.

7.5.1 Gadgets

In our reduction, both variable and clause gadgets are plane oriented graphs, and eventually, none will be embedded in an internal (non-external) face of another. Therefore, while the outerface of any gadget may interact with the rest of the instance, logical choices for variables are encoded by *internal* completions, where all arcs are embedded in its inner faces. Inner faces are either triangles, in which no arc can be added, or contain internal terminals (not incident to the outerface), restricting valid completions.

Given two completions Y, X of H, we say Y *dominates* X if the partition of $H + X$ into strong components is a strict refinement of the one of $H + Y$. If two completions admit the same partition, they are *equivalent*. Then, a completion is *maximal* when it is not dominated by another. A completion X of H is *valid* if it is internal, and all terminals components of $H + X$ lie on the outerface. We will (implicitly) use the following throughout this section:

Observation 7.5.1. *Let D be a positive instance for PSCA, with solution Y, and H be a (plane) subidgraph of D. Then, the restriction Y_H of Y to H is valid, and replacing Y_H in Y with a maximal valid completion still yields a solution.*

Clauses

Consider a clause $\phi = a \vee b \vee c$ of Φ, where a, b, c are literals corresponding to variables x, y, z, and let us build the corresponding gadget, depicted in Figure 7.5. We start with three vertices v_a, v_b, v_c, and add an antidirected path of length 4 between each pair as follows. The path from v_a to v_b induces vertices $(v_a, v_{aab}, v_{ab}, v_{abb}, v_b)$, where v_a, v_{ab}, v_b are sinks, and v_{aab}, v_{abb} are sources. The paths from v_b to v_c and from v_c to v_a are in the same way, with corresponding internal vertices labelled accordingly. At this point, we have built an antidirected cycle of length 12, which we embed as plane, distinguishing the inner and outer faces. We add a vertex s on the inside faces, with in-arcs from v_a, v_b, v_c, and out-arcs towards all other vertices. This achieves to build the gadget for ϕ, consisting in sinks v_{ab}, v_{bc}, v_{ca}, and a source component containing all other vertices. Informally, vertices v_a, v_b, v_c now correspond to "entry points" to the clause gadget, which may only be attained from the rest of the instance if the variable gadgets are completed accordingly. We note that arcs towards s can also be removed to yield an acyclic clause gadget, but their presence facilitates our proofs.

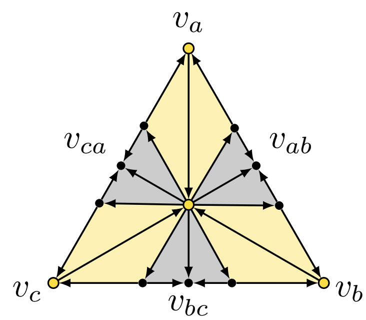


Figure 7.5: A clause gadget C_ϕ, with its (unique) source component S_ϕ in yellow. Vertices v_a, v_b, v_c correspond to literals, later identified to vertices of the corresponding variable gadgets. Then, arcs will be added from the middle local sinks v_{ab}, v_{bc}, v_{ca} to ensure C_ϕ contains no sinks.

Literals

Variable gadgets are constructed from as many literal gadgets as there are occurences in the formula. Let us describe the construction of a literal gadget, depicted in Figure 7.6. We start with an alternating cycle of length 8, embedded planarily. We choose an arbitrary sink to be the bottom sink and label it b, then label remaining sinks in clockwise order as l, t, r for "left", "top" and "right". We label the sources in-between two sinks as bl, tl, tr and br according to the sinks they are adjacent to. Create twins b', l', t', r' for each sink, and embed them on the outer face of the octagon, then, add an arc from the twin to the original sink. Then, add the arc (t, b) inside the octagon, leaving only b, l, r as sinks. This results in two 5-faces, which are the only ones that may be completed.

We say that a completion X of a gadget *hits* a source s if it contains an arc towards s, then X hits a sink t if it contains an arc from t. Sinks of a literal gadget are internal to it, and they will remain internal in the variable gadget, forcing them all to be hit by a valid completion. In the variable gadget, bottom sources of each literal will also be internal, which will force at least

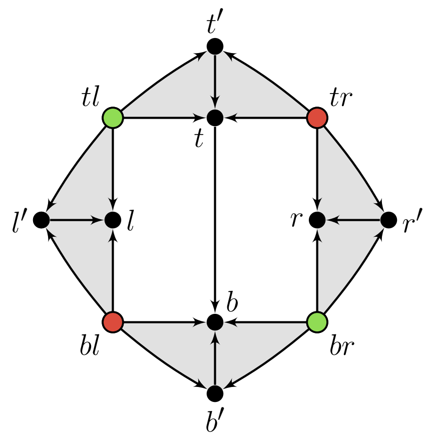


Figure 7.6: A literal gadget L, with internal sinks b, l, r, bottom sources bl, br, and top sources tl, tr.

one to be hit. The following lemma shows that these two constraints are enough to restrict our completions to exactly two maximal ones.

Lemma 7.5.2. *Any valid completion of the literal gadget L hitting br is either:*

- *The* positive completion $Y = \{(r, t), (t, br), (l, b), (b, tl)\}$*, yielding sources bl, tr in $L + Y$; or*
- $Y = \{(r, t), (t, br), (b, l), (l, t)\}$*, yielding sources bl, tr, tl in $L+Y$, and dominated by the positive completion.*

Any valid completion hitting bl is either:

- *The* negative completion $Y = \{(l, t), (t, bl), (r, b), (b, tr)\}$*, yielding sources br, tl in $L + Y$; or*
- $Y = \{(l, t), (t, bl), (b, r), (r, t)\}$*, yielding sources br, tr, tl in $L + Y$, and dominated by the negative completion.*

In all cases, all non-sources vertices form a sink component.

Proof. Let Y be any valid completion hitting at least one of the bottom sources bl and br. Since l, b, r are internal sinks, they must be hit by Y, and observe that no arc hitting those can be added towards bl or br. This means $|Y| = 4$ and exactly two arcs of Y are embedded in each of the two faces incident to the arc (t, b).

Assume first that Y hits br. The arc hitting br is exactly $(t, br) \in Y$, as otherwise the internal sink r could not be hit. Thus, to hit r, the (unique) other arc of Y in the right 5-face is (r, t). If the arc of Y hitting b is (b, tl), this yields the positive completion $\{(r, t), (t, br), (l, b), (b, tl)\}$ (see the second completion in Figure 7.7). Otherwise, the arc of Y hitting b must be (b, l), which yields completion $\{(r, t), (t, br), (b, l), (l, t)\}$. In both cases, since bl, tr are not hit, they are sources of $L + Y$, and in the second case, tl must also be. In either case, all non-source vertices form a sink component.

The case is symmetrical if Y hits bl, which forces arcs of Y added in the left case to be (t, bl) and (l, t). Then, according to the arc hitting b in the left case, we either obtain the negative completion $Y = \{(l, t), (t, bl), (r, b), (b, tr)\}$, yielding sources br, tl (third completion in Figure 7.7). Or,

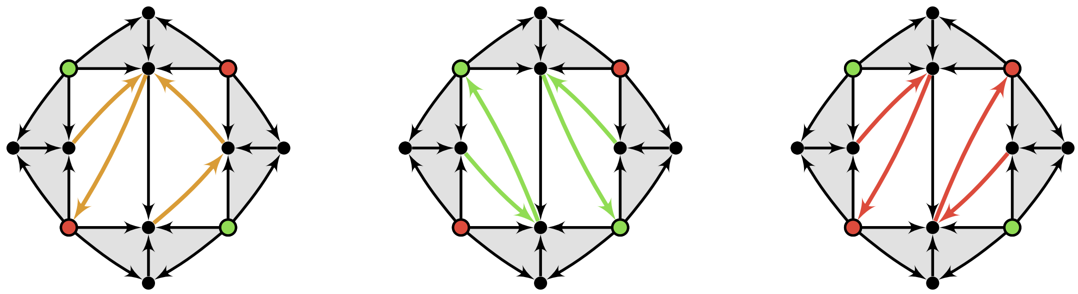


Figure 7.7: A literal gadget L along with three completions. The first completion is valid and hits a bottom source, but is dominated by the third. The second and third completions correspond to the positive and negative (valid) completions respectively. These hit exactly one bottom source, and form a sink component containing all vertices but the two antipodal sources that are not hit.

we obtain $Y = \{(l,t),(t,bl),(b,r),(r,t)\}$, yielding sources br, tl, tr (first completion in Figure 7.7). □

Variables

Let us formally describe the construction of the gadget G_x for a variable x appearing n_x times in ϕ. Note that we may assume $n_x \geq 2$ as otherwise the 3-SAT instance can be reduced by removing G_x and the associated clause. Take n_x copies of a literal gadget, $L_1, ..., L_{n_x}$, with corresponding vertices now labelled with subscripts $i \in [1, n_x]$. Considering indices i modulo n_x in $[1, n_x]$, we build the variable gadget for x as follows:

- for $i \in [1, n_x]$, identify sources $br_i \in V(L_i)$ and $bl_{i+1} \in V(L_{i+1})$,
- for $i \in [1, n_x]$, identify r'_i with l'_{i+1}.
- Finally, identify all $(b'_i)_i$ into a common vertex b'.

We embed the resulting gadget canonically, such that the (antidirected) cycle formed by $(br_i)_i$ is plane, and the embedding of any L_i is preserved. The final construction is depicted in Subsection 7.5.1 with a completion which will correspond to a positive assignment.

Lemma 7.5.3. *A variable gadget G_x admits exactly two maximal valid completions:*

- *The* positive completion*, with source vertices $(tr_i)_i$ and sink component $Y_x = V(G_x)\setminus\{tr_i : i\}$.*
- *The* negative completion*, with source vertices $(tl_i)_i$ and sink component $Y_x = V(G_x)\setminus\{tl_i : i\}$.*

Proof. Consider a variable gadget G_x, comprised of literal gadgets $(L_i)_i$ for $i \in [1, n_x]$. Let Y be a maximal valid completion of G_x, and $Y_i = Y \cap V(L_i)^2$ be its restriction to literal gadget L_i.

The internal sinks of G_x are exactly $(l_i, b_i, r_i)_i$, for $i \in [n_x]$, meaning each Y_i is a valid completion of L_i. The internal sources of G_x are exactly $(br_i)_i$, which must all be hit by arcs of Y inside the literal gadgets, by construction. Then, we know from Lemma 7.5.2 that at most one such arc may belong to a single Y_i. So each Y_i contains exactly one arc incident to br_i or $bl_i = br_{i-1}$. We claim Y_i must also be maximal for L_i. Assume it is not, then it must be one of the two completions

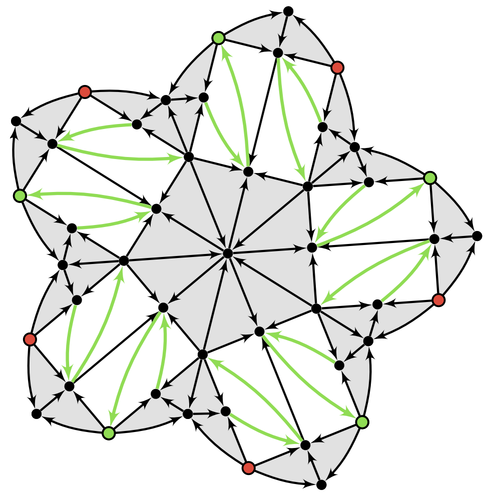


Figure 7.8: The gadget G_x for variable x, appearing in 5 clauses, with a maximal completion corresponding to the positive assignment shown in green. This yields a valid completion with top-right vertices of each literal being sources, while all other vertices form a sink component.

dominated by the positive and negative ones, thanks to Lemma 7.5.2. In both cases, tl_i and tr_i are sources in $L_i + Y_i$, thus also in $G_x + Y$. Then, substituting Y_i with the completion dominating it would yield a completion of G_x dominating Y, which contradicts its maximality. Therefore, each Y_i is either the positive or the negative completion of L_i.

Now, assume for some i that Y_i is the positive completion of L_i, that is, the (unique) valid completion hitting br_i. Since bl_i and br_{i-1} are identified, and because bl_i is a source in $L_i + Y_i$, bl_i must now be hit by Y_{i-1}. This implies that Y_{i-1} is the positive completion of L_{i-1} as well. Applying the above inductively yields that each Y_j is exactly the positive completion of L_j for all $j \in [n_x]$. Conversely, if for some i, the completion Y_i is the negative completion of L_i, then this is the case for all i.

In turn, Y must be one these two completions, which we call *positive* and *negative* respectively. In both cases, Y contains a directed cycle containing all internal sources of G_x. If Y is positive, it can be obtained "clockwise" by concatenating directed paths $(bl_i, b_i, tl_i, t_i, bl_{i+1})$. If Y is negative, it can be found "anti-clockwise" by concatenating directed paths $(br_i, b_i, tr_i, t_i, bl_{i-1})$. Applying Lemma 7.5.2, $L_i + Y_i$ consists in a sink component containing all but source vertices $\{bl_i, tr_i\}$ in the positive case, or $\{br_i, tl_i\}$ in the negative case. Combining the two observation, $G_x + Y$ contains sources $\{tr_i : i\}$ and sink component $Y_x = V(G_x) \setminus \{tr_i : i\}$ in the positive case. And in the negative case, it contains sources $\{tl_i : i\}$ and sink component $Y_x = V(G_x) \setminus \{tl_i : i\}$. This shows that both completions are indeed valid, and the unique such ones that are maximal. □

7.5.2 Construction

Consider now a Plane 3-SAT instance $\Phi = \bigwedge_{j \in [m]} \phi_j$ on variables $(x_i)_{i \in n}$, with an embedding for which the cycle $(x_1, ..., x_n)$ may be added while preserving planarity. We describe a plane oriented graph D which will be an equivalent instance for PSCA.

We embed variable and clause gadgets corresponding to Φ canonically, respecting the embedding of Φ. For each variable x, consider the corresponding gadget G_x formed by its n_x literals. We label each literal gadget as $L_{x,\phi}$ for any ϕ containing x. That is, the (clockwise) order of literal gadgets inside G_x follows the order of incidences of (literals of) x in the corresponding clauses of Φ. Likewise, when writing $\phi = a \vee b \vee c$, we assume a, b, c are ordered clockwise according to their incidence on the corresponding variables in Φ.

Let us start by describing vertex/clause incidences. Consider any variable x involved in some clause $\phi = a \vee l(x) \vee c$. Taking $L_{x,\phi}$ to be the literal gadget of x associated to ϕ, and C to be the clause gadget of ϕ, we join both gadgets as follows.

- If $l(x) = x$, we identify vertices $v_x \in V(C)$ and $tl_{x,\phi} \in V(L_{x,\phi})$. Then, add arcs $(v_{ax}, l'_{x,\phi})$, $(v_{axx}, l'_{x,\phi})$ as well as $(v_{xc}, tr_{x,\phi}), (v_{xc}, t'_{x,\phi}), (v_{xxc}, t'_{x,\phi})$.
- If $l(x) = \overline{x}$, we identify vertices $v_{\overline{x}} \in V(C)$ and $tr_{x,\phi} \in V(L_{x,\phi})$. Then, add arcs $(v_{xc}, r'_{x,\phi})$, $(v_{xxc}, r'_{x,\phi})$ as well as $(v_{ax}, tl_{x,\phi}), (v_{ax}, t'_{x,\phi}), (v_{axx}, t'_{x,\phi})$.

We apply the operation above to any variable/clause incidence, which can be done while preserving the planar embedding. Note we must ensure that for each $\phi = a \vee b \vee c$, the variable/literal incidence respects that of Φ. That is, when our definition considers $\phi = a \vee b \vee c$ to join literal (gadget) b, we then consider $\phi = c \vee a \vee b$ to join a, and $\phi = b \vee c \vee a$ to join c. See Figure 7.9 for the resulting subdigraph of a clause joined with its three variables.

Then, it remains to connect the variable gadgets following the variable cycle $(x_1, ..., x_n)$. Consider the edge (x_i, x_{i+1}) in the variable cycle of Φ, and take G_i and G_{i+1} to be the variable gadgets of x_i and x_{i+1}. Then, by our construction and the assumption on the embedding of Φ, we can safely add an arc between G_i and G_{i+1}, between vertices other than top sources. Indeed, it suffices to take two vertices of type l' in appropriate literal gadgets of x_i and x_{i+1}, and add an arc from the first to the second. This achieves the construction of the reduced instance.

7.5.3 Soundness

We are now ready to show the hardness of Plane Strong Connectivity Augmentation.

Theorem 7.1.1. *Plane Strong Connectivity Augmentation is NP-complete.*

Proof. Consider a plane 3-SAT instance Φ on variables $(x_i)_i$, and let D be the construction obtained from Φ described in Subsection 7.5.2 as the instance of PSCA.

The initial instance is positive implies the reduced instance is.

Assume first that Φ is a positive instance for 3-SAT, and consider a a boolean assignment for $(x_i)_i$ satisfying it. Now, for any i, if x_i is set to true, we let X_i be the (embedded) positive completion of variable gadget G_{x_i}, and otherwise let X_i be its negative completion, as defined in Lemma 7.5.3. Then, we let $X = \bigcup_i X_i$, with the embedding respecting those of each X_i in G_{X_i}. By construction, and since all X_i are internal completions of different gadgets, $D + X$ is plane and oriented. To show that it is indeed a solution for D, it remains to prove strong connectivity.

Take any variable x_i and corresponding gadget G_{x_i}. Then, Lemma 7.5.3 ensures that the sources of $G_{x_i} + X_i$ are exactly $(tr_{i,j})_j$, or exactly $(tl_{i,j})_j$, abusing notation to let j index clauses containing x_i clockwise. Moreover, all non-source vertices of G_{x_i} form a single sink component Y_{x_i}. By construction, for each i modulo n in $[1, n]$, our construction adds an arc from some $l'_{i,j}$ to $l'_{i+1,q}$

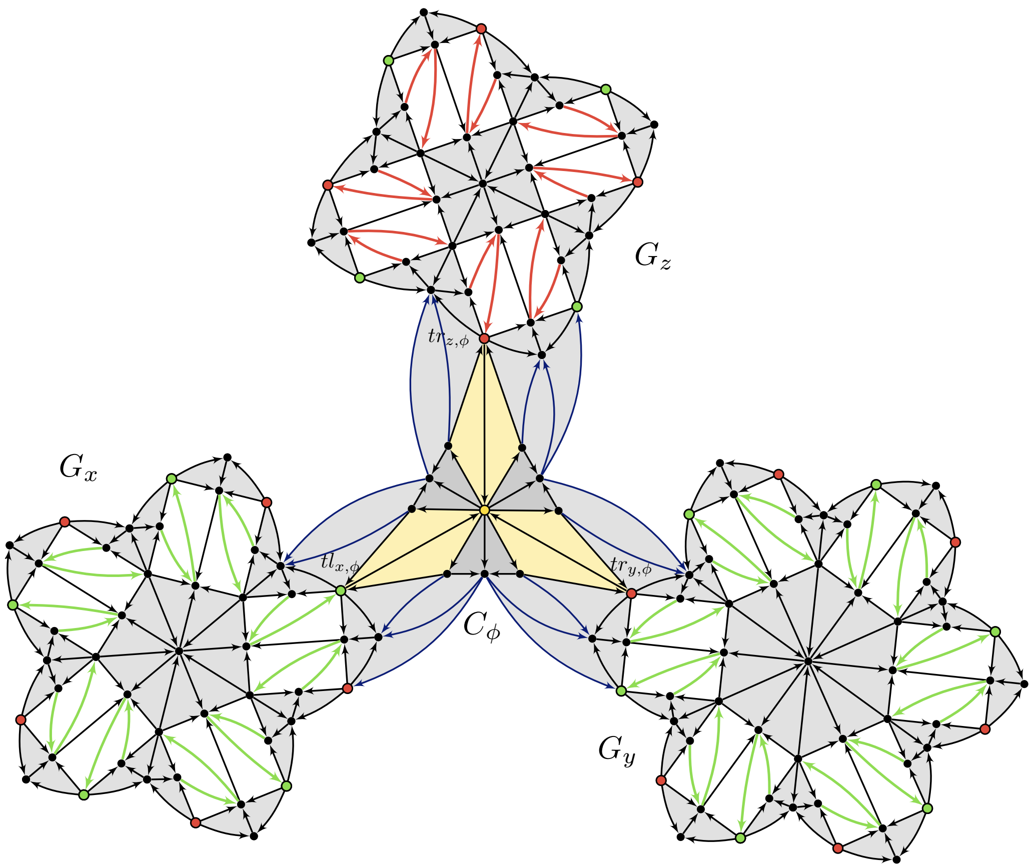


Figure 7.9: A clause gadget C_ϕ for $\phi = x \vee \overline{y} \vee \overline{z}$. Gadgets G_x, G_y and G_z correspond to variables x, y and z, appearing 5, 6, and 4 times in Φ respectively. A top source of each literal gadget is now identified to the source component of the clause. Here, x and z are completed as to satisfy ϕ, allowing the clause gadget to be entered from variables x and z, but not from y.

for some j, q. In particular, these are arcs from Y_{x_i} to $Y_{x_{i+1}}$, yielding that Y_{x_i} belong to the same strongly connected component, say Y, in $D + X$. We also know that, in D, arcs have been added from the all clause gadget sinks to some $l'_{i,j}$ in variable gadgets, see the blue arcs in Figure 7.9. In particular, any initial sink (component) of D belongs to some Y_i. Now, since any sink component of $D + X$ must contain a sink of D, it must intersect some Y_i. Therefore, the only possible sink component of $D + X$ is Y, possibly with $Y = V(D)$.

Since D is connected, to obtain strong connectivity of $D+X$, it suffices to show that any source component is also Y. If S is a source component of $D + X$, it must contain at least one source component of D. By construction, the source components of D are the sources S_ϕ of gadgets C_ϕ. Take then some ϕ such that $S_\phi \subseteq S$, say with $\phi = l(x) \vee l(y) \vee l(z)$. Recall then that vertices $v_{l(x)}, v_{l(y)}, v_{l(z)} \in S_\phi$ have each been identified with either some corresponding $tl_{\_,\phi}$ or $tr_{\_,\phi}$ in gadgets G_x, G_y, G_z. Assume without loss of generality that our boolean assignment is such that ϕ is satisfied by x. If $l(x) = x$, we have $tl_{x,\phi} = v_{l(x)} \in S_\phi$, and since X_i is the positive completion of G_x, $tl_{x,\phi} \in Y$, meaning $S_\phi \subseteq Y$. The argument is symmetric when $l(x) = \bar{x}$, as $tr_{x,\phi} \in Y \cap S_\phi$. In any case $S_\phi \subseteq Y$, and therefore $S \subseteq Y$, meaning $D + X$ is strongly connected. See Figure 7.9, where Y contains $tl_{x,\phi}$ and thus S_ϕ as x appears positively.

The reduced instance is positive implies the initial instance is.

Conversely, assume D is a positive instance of PSCA, take X to be a completion achieving strong connectivity. Without loss of generality, we can take X to be maximal within each variable gadget $G_i = G_{x_i}$, and let $X_i = X \cap V(G_i)$. Since X_i is a maximal completion of G_i, which is valid by strong connectivity of $D+X$, Lemma 7.5.3 implies X_i is either a positive or negative completion. Then, we define our boolean assignment for $(x_i)_i$ according to the type of completion of X_i, and show it satisfies Φ.

Towards this, consider any clause $\phi = l(x) \vee l(y) \vee l(z)$, corresponding gadget C_ϕ, and source component S_ϕ of C_ϕ. Since $D + X$ is strongly connected, X must contain an arc towards some vertex of S_ϕ. Since this arc does not create a digon, it cannot be embedded in a triangle of D. By construction, the arcs between C_ϕ and each variable gadget G_x, G_y, G_z ensure that in D, the only vertices of S_ϕ which are incident to faces other than triangles are $v_{l(x)}, v_{l(y)}, v_{l(z)}$. Now, these are each identified with a top source tl or tr of the corresponding variable gadgets. More precisely, each $v_{l(x)}, v_{l(y)}, v_{l(z)}$ is incident to exactly one non-triangle face in D, which is one of the 5-faces of the corresponding literal gadgets. Assume without loss of generality that an arc of X towards S_ϕ is incident to $v_{l(x)}$, and is embedded in a 5-face of literal $L = L_{x,\phi}$. We know that the restriction X_L of X to $L_{x,\phi}$ is either its positive or negative completion, according to the completion of G_x. If $l(x) = x$, by construction we have $v_{l(x)} = tl_{x,\phi}$. Then, Lemma 7.5.2 yields that X_L is the positive completion $L_{x,\phi}$. Therefore, G_x has been completed positively, meaning we have assigned x to true, and ϕ is indeed satisfied. The case is symmetrical when x appears negatively in ϕ, meaning $v_{l(x)} = tr_{x,\phi}$: X_L is the negative completion of L, meaning our false assignment to x satisfies ϕ. Considering this for any clause of Φ achieves to show that our boolean assignment to $(x_i)_i$ satisfies Φ. $\square$

7.6 Further directions

Our results leave two main open problems, the most important one being whether DIRECTED-PSCA admits a polynomial-time algorithm, which we conjecture.

Conjecture 7.6.1. *There exists a polynomial-time algorithm for DIRECTED-PSCA.*

Another important question is the fixed-parameter tractability of PSCA when parameterized by the number of terminals. We conjecture that unlike STRONGLY CONNECTED STEINER SUBGRAPH (see Theorem 6.1.23), this problem should not be $W[1]$-hard.

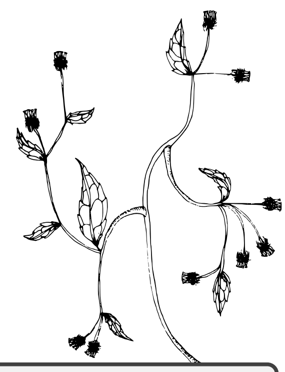
Conjecture 7.6.2. *PLANE STRONG CONNECTIVITY AUGMENTATION is FPT with respect to the number of terminals.*

We recall that the fixed-parameter tractability for connectivity augmentation in planar (or plane) undirected graphs remains open. In particular, we lack an undirected analogue of Theorem 7.1.2 for biconnectivity.

Question 7.6.3. *Is PLANE 2-VCA FPT parameterized by solution size?*

Chapter 8

Inversions of bounded size



This chapter investigates inversions of bounded size towards acyclicity, as introduced in Section 6.2. It is based on joint work with Jørgen Bang-Jensen, Frédéric Havet, Florian Hoersch, Clément Rambaud and Caroline Silva, appearing in [Ban+25].

8.1 Introduction

This chapter deals with inversions of bounded or prescribed size, namely $(\leq p)$-*inversions* and $(=p)$-*inversions*. Recall these are the operations operating over vertex sets of size at most p, respectively exactly p, reversing the direction of all induced arcs. As we have seen in Subsection 6.2.5, most of the literature deals with inversions of arbitrary size. Bounded (or prescribed) size inversions sit in between general inversions and arc reversals, which correspond to the case $p = 2$. Then, for a digraph $\overrightarrow{G}$, the $(=p)$-*inversion number* $\text{inv}^{=p}(\overrightarrow{G})$, and the $(\leq p)$-*inversion number* $\text{inv}^{\leq p}(\overrightarrow{G})$, are the minimum number of corresponding operations rendering $\overrightarrow{G}$ acyclic. Clearly $\text{inv}^{=p}(\overrightarrow{G}) \geq \text{inv}^{\leq p}(\overrightarrow{G})$ and $\text{inv}^{=2}(\overrightarrow{G}) = \text{inv}^{\leq 2}(\overrightarrow{G}) = \text{fas}(\overrightarrow{G})$.

While ($\leq p$)-inversions were already studied by Yuster [Yus25], who showed extremal bounds on $\text{inv}^{\leq p}(\overrightarrow{G})$, we initiate the study of ($= p$)-inversions both algortihmically and extremally. Then, our (parameterized) complexity results deal with both ($= p$)-inversions and ($\leq p$)-inversions.

We begin our study of ($= p$)-inversions in Section 8.3, through the algorithmic problem of ($= p$)-invertibility in oriented graphs. There, rather than asking for a minimum number of inversions, we only wish to decide whether $\text{inv}^{=p}(\overrightarrow{G})$ is finite. First, we prove that deciding whether an oriented graph of order n is ($= n - 1$)-invertible is an NP-complete problem.

Theorem 8.1.1. Vertex-invertibility *is NP-complete.*

Next, we show that this is essentially the only case when the ($= p$)-invertibility is hard.

Theorem 8.1.2. *There is a polynomial-time algorithm that, given a positive integer p and an oriented graph D of order distinct from $p + 1$, decides whether D is ($= p$)-invertible.*

In order to prove this theorem, we first characterize in Subsection 8.3.2 the ($= p$)-invertible tournaments of order at least $p + 2$. Then, in Subsection 8.3.3, we use this characterization to give necessary and sufficient conditions for an oriented graph to be ($= p$)-invertible. In particular, we show that if p is even, then every oriented graph of order at least $p + 2$ is ($= p$)-invertible (Corollary 8.3.14). Using those conditions, we describe the above-mentioned polynomial-time algorithm.

In Section 8.4, we investigate upper bounds on the ($= p$)-inversion number in terms of the feedback-arc-set number or the ($\leq p$)-inversion number. We identify a dichotomy between even and odd values of p : when p is even the ($= p$)-inversion number is at most $4(p - 1)\binom{p}{2}$ times the ($\leq p$)-inversion number (Corollary 8.4.5), while there is no upper bound on the ($= p$)-inversion number as a function of the ($\leq p$)-inversion number when p is odd (Proposition 8.4.6).

Finally, in Section 8.5, we study the complexity of deciding the ($\leq p$)-inversion number and ($= p$)-inversion number of a given oriented graph.

($\leq p$)-Inversion
Input: An oriented graph D and a positive integer k
Question: Does D satisfy $\text{inv}^{\leq p}(D) \leq k$?

($= p$)-Inversion
Input: An oriented graph D and a positive integer k
Question: Does D satisfy $\text{inv}^{=p}(D) \leq k$?

Note that when $p = 2$, both problems correspond to computing the feedback-arc-set number of an oriented graph, which is NP-hard [Kar72]. In Subsection 8.5.1, we show that both problems are also NP-hard for any fixed p greater than 2, even when restricted to tournaments.

Theorem 8.1.3. Tournament ($\leq p$)-Inversion *and* Tournament ($= p$)-Inversion *are NP-complete, for any fixed integer p greater than* 2.

In Subsection 8.5.2, we prove the following kernelization when the input is a tournament.

Theorem 8.1.4 (see Theorem 8.5.4)**.** *When considering p as an input,* Tournament ($\leq p$)-Inversion *admits a $(1 + \epsilon)k^2p^3 + o(k^2p^3)$ kernel for every $\epsilon > 0$.*

This implies that the problem is FPT when parameterized by k and p. In contrast, we show that for general oriented graphs, the problem is hard when parameterized by p even when $k = 1$.

Theorem 8.1.5. *Given a digraph D, both deciding whether D can be made acyclic by one $(\leq p)$-inversion, and deciding whether D can be made acyclic by one $(= p)$-inversion, are $W[1]$-hard parameterized by p.*

8.2 Preliminaries

A $(= p)$-*set* (resp. $(\leq p)$-*set*) is a set of cardinality exactly p (resp. at most p). Let D be a digraph and $\mathcal{X}$ a family of subsets of $V(D)$. We say that $\mathcal{X}$ is a $(= p)$-*family* (resp. $(\leq p)$-*family*) if all members of $\mathcal{X}$ are $(= p)$-sets (resp. $(\leq p)$-sets). We denote by $\text{Inv}(D; \mathcal{X})$ the digraph obtained after inverting all sets of $\mathcal{X}$ one after another. Observe that this is independent of the order in which we invert those sets: $\text{Inv}(D; \mathcal{X})$ is obtained from D by reversing exactly those arcs for which an odd number of members of $\mathcal{X}$ contain both endvertices. If $\mathcal{X} = \{X\}$, then we write $\text{Inv}(D, X)$ for $\text{Inv}(D; \mathcal{X})$. A *decycling family* of a digraph D is a family $\mathcal{X}$ of subsets of $V(D)$ such that $\text{Inv}(D; \mathcal{X})$ is acyclic.

For a digraph D, let $\sigma = (v_1, v_2, \dots, v_n)$ be an ordering of the vertices of D. An arc (v_i, v_j) is *forward* (according to σ) if $i < j$ and *backward* (according to σ) if $j < i$. It is well known that a digraph is acyclic if, and only if, it admits an *acyclic ordering*. That is, an ordering of its vertices with no backward arcs. A *median order* of a digraph D is an ordering of the vertices of D with the maximum number of forward arcs, or, equivalently, the minimum number of backward arcs. Note that this is also an ordering for which the arcs of a minimum feedback arc set are all backward.

8.3 $(= p)$-invertibility

8.3.1 NP-completeness of $(n-1)$-Invertibility

Let D be a digraph. The *vertex-inversion* of a vertex v in D consists in reversing all the arcs incident to v. If X is a set of vertices, then we denote by $\text{VInv}(D, X)$ the digraph obtained after the vertex-inversions of all vertices of X. Note that this is independent of the order of the vertex-inversions. An arc is reversed if and only if exactly one of its end-vertices is in X. A digraph is *vertex-invertible* if there is a sequence of vertex-inversions that makes D acyclic, that is if there exists a set X of vertices such that $\text{VInv}(D, X)$ is acyclic. Such a set X is called a *vertex-inversion set*. Observe that $\text{VInv}(D, \{v\})$ is the converse of $\text{Inv}(D, V(D) \setminus \{x\})$.

Observation 8.3.1. *A digraph on n vertices is vertex-invertible if and only if it is $(= n-1)$-invertible.*

We denote by Vertex-invertibility the problem of deciding whether a given digraph is vertex-invertible. By Observation 8.3.1, Vertex-invertibility is equivalent to the problem $(n-1)$-Invertibility of deciding whether a digraph of order n is $(= n-1)$-invertible. Therefore, proving the NP-completeness of $(n-1)$-Invertibility is equivalent to prove the NP-completeness of Vertex-invertibility. We prove it in the remainder of this subsection. We first establish some preliminary results.

The vertex-inversions of all vertices of a set X results in inverting all the arcs between X and $V(D) \setminus X$, so applying vertex-inversions to all vertices of X is equivalent to applying them on all vertices of $V(D) \setminus X$.

Observation 8.3.2. *If X is a vertex-inversion set of D, then $V(D) \setminus X$ is also a vertex-inversion set.*

Lemma 8.3.3. *Let D be an acyclic digraph and v a vertex of D. One can invert a set X of vertices such that* $\text{VInv}(D, X)$ *is acyclic with source v.*

Proof. Consider an acyclic ordering $(v_1, \dots, v_n)$ of D. Let i be the integer such that $v_i = v$. Set $X = \{v_1, \dots, v_{i-1}\}$. One can easily check that $\text{VInv}(D, X)$ is acyclic and that v is a source in this digraph. □

Corollary 8.3.4. *Let D be a vertex-invertible digraph. Then, for every $v \in V(D)$, there exists $X \subseteq V(D)$ such that* $\text{VInv}(D, X)$ *is an acyclic digraph with source v.*

Corollary 8.3.5. Vertex-invertibility *is polynomial-time solvable for tournaments.*

Proof. Let T be a tournament and let $v \in V(T)$. By Corollary 8.3.4, it suffices to check if $\text{VInv}(T, N^-(v))$ is an acyclic digraph. □

In general digraphs, we show a reduction from 2-Dicolourability, which is known to be NP-complete due to Bokal [Bok+04].

Theorem 8.1.1. Vertex-invertibility *is NP-complete.*

Proof. We prove the statement by a reduction from 2-Dicolourability. Let D be a digraph. Let D' be the digraph obtained from D as follows.

- Subdivide each arc of D once and let S be the set of subdivided vertices.
- Add a new vertex v and the arc (v, u) for all $u \in S$.

Now we prove that D is a positive instance for 2-Dicolourability if and only if D' is a positive instance for Vertex-invertibility.

Suppose first that D is a positive instance for 2-Dicolourability. Let $\{X_1, X_2\}$ be a 2-dicolouring of D. We claim that $H = \text{VInv}(D', X_1)$ is acyclic. Towards a contradiction, suppose the opposite and let C be a directed cycle of H. Note that v is a source in H, so $v \notin V(C)$. So the vertices of C must alternate between S and $X_1 \cup X_2$. Moreover, note that if $u \in S$ and u has a neighbour in both X_1 and X_2, then u is either a source or a sink in H. So $V(C) \subseteq X_1$ or $V(C) \subseteq X_2$. This implies that $D[X_1]$ or $D[X_2]$ is not acyclic, a contradiction.

Suppose now that D' is a positive instance for Vertex-invertibility. Let X be a vertex-inversion set of D'. By Observation 8.3.2 and Corollary 8.3.4, we may assume that $v \notin X$ and that v is the source of $\text{VInv}(D', X)$. Since v is also a source of D' and $N^+_{D'}(v) = S$, it follows that $X \cap S = \emptyset$, so $X \subseteq V(D)$. We claim that $\{X, V(D) \setminus X\}$ is a 2-dicolouring of D. Towards a contradiction, suppose the opposite. Suppose first that $D[X]$ has a directed cycle $(u_1, \dots, u_\ell, u_1)$. This implies that $(u_\ell, w_\ell, u_{\ell-1}, w_{\ell-1}, \dots, u_1, w_1, u_\ell)$ is a directed cycle of $\text{VInv}(D', X)$ where w_i is the subdivided vertex of (u_i, u_{i-1}), a contradiction. Suppose now that $D - X$ has a directed cycle $(u_1, \dots, u_\ell, u_1)$. This implies that $(u_1, w_1, u_2, w_2, \dots, u_\ell, w_\ell, u_1)$ is a directed cycle of $\text{VInv}(D', X)$, where w_i is the subdivided vertex of (u_i, u_{i+1}), a contradiction. □

8.3.2 Characterization of $(=p)$-invertible tournaments

The aim of this section is to prove the following theorem.

Theorem 8.3.6. *Given an integer $p \geq 2$ and two orientations $\overrightarrow{G}_1, \overrightarrow{G}_2$ of the same graph on $n \geq p+2$ vertices, one can decide in linear time whether there exists a $(=p)$-family $\mathcal{X}$ of subsets of $V(G)$ such that* $\text{Inv}(\overrightarrow{G}_1; \mathcal{X}) = \overrightarrow{G}_2$.

Let p, n be integers with $p \geq 2$ and $n \geq p+2$. When we want to invert only sets of size exactly p, several obstructions appear, depending on the value of p modulo 4. Let T be a tournament with vertex set $[n]$. Observe that when p is odd, for any vertex v, $d^+(v) \bmod 2$ is invariant under $(=p)$-inversions. Moreover, when $\binom{p}{2}$ is even (that is when $p \equiv 0 \bmod 4$ or $p \equiv 1 \bmod 4$), the parity of the number of backward arcs with respect to the natural ordering (the arcs ji with $i < j$) is also invariant under $(=p)$-inversions. We will actually show that these two invariants are essentially the only ones, which gives a characterization of tournaments reachable from any given tournament via $(=p)$-inversions. To prove that, we will use an algebraic representation of the problem.

We shall do all operations in the two-element field $\mathbb{F}_2$. For convenience, we index the vectors of the vector space $\mathbb{F}_2^{\binom{n}{2}}$ by the $(=2)$-sets of $[n]$. Thus, the coordinates of a vector $\mathbf{u} \in \mathbb{F}_2^{\binom{n}{2}}$ are indexed by $(=2)$-sets of $[n]$, and element of $\mathbf{u}$ are given by $\mathbf{u}_{\{i,j\}}$ for $i, j \in [n]^2$.

We start by defining a representation of (labelled) n-vertex tournaments by $\binom{n}{2}$-dimensional vectors over $\mathbb{F}_2$. Let T be a tournament with vertex set $[n]$. We define $\mathbf{a}_T \in \mathbb{F}_2^{\binom{n}{2}}$ by

$$(\mathbf{a}_T)_{\{i,j\}} = \begin{cases} 1 & \text{if } ji \in A(T), \\ 0 & \text{if } ij \in A(T), \end{cases}$$

for every $i, j \in [n]$ with $i < j$. Let $X \subseteq [n]$, and let $\mathbf{i}_X \in \mathbb{F}_2^{\binom{n}{2}}$ be defined by

$$(\mathbf{i}_X)_{\{i,j\}} = \begin{cases} 1 & \text{if } i, j \in X, \\ 0 & \text{otherwise}, \end{cases}$$

for every $i, j \in [n]$ with $i \neq j$. The point of these definitions is the following observation.

Observation 8.3.7. *For every tournament T on $[n]$ and for every $X \subseteq [n]$,*

$$\mathbf{a}_{\text{Inv}(T;X)} = \mathbf{a}_T + \mathbf{i}_X.$$

The proof of Theorem 8.3.6 will consist in defining a linear mapping $\pi \colon \mathbb{F}_2^{\binom{n}{2}} \to \mathbb{F}_2^t$ for some nonnegative integer t such that for all tournaments T', T on $[n]$,

$$\begin{gathered} \text{there exists a } (=p)\text{-family } \mathcal{X} \subseteq 2^{[n]} \text{ such that } \text{Inv}(T;\mathcal{X}) = T' \\ \text{if and only if } \pi(\mathbf{a}_T) = \pi(\mathbf{a}_{T'}). \end{gathered} \tag{8.1}$$

Informally, π corresponds here to the most general quantity which is invariant under $(=p)$-inversions.

The property (8.1) can be reformulated as follows. For every $\mathbf{u} \in \mathbb{F}_2^{\binom{n}{2}}$, with $\mathbf{0}$ being the null element of $\mathbb{F}_2^t$:

$$\mathbf{u} \in \text{Span}\{\mathbf{i}_X \mid X \subseteq [n], |X| = p\} \text{ if and only if } \pi(\mathbf{u}) = \mathbf{0}. \tag{8.2}$$

It will be clear from the definition of π that it can be computed in linear time, and so Theorem 8.3.6 will follow. The following facts will be used several times in the construction of π.

Claim 8.3.1. With four $(=p)$-inversions, we can reverse the arcs of a 4-cycle and no other arcs.

Proof of C. onsider four vertices x_1, x_2, x_3, x_4 and X a set of $p-2$ other vertices. Let us invert the sets $X \cup \{x_1, x_2\}$, $X \cup \{x_2, x_3\}$, $X \cup \{x_3, x_4\}$, and $X \cup \{x_4, x_1\}$. Doing so, all arcs are reversed an even number of times except those of the 4-cycle $(x_1, x_2, x_3, x_4, x_1)$. ◇

Claim 8.3.2. Assume p is even. With $2(p-1)$ $(=p)$-inversions, we can reverse any two adjacent arcs and no others.

Proof of C. onsider three vertices u, v_1, v_2 and let us show how to reverse the arcs between u and v_1, and u and v_2. Let X be any set of $p-2$ other vertices. Invert first $X \cup \{u, v_1\}$ and $X \cup \{u, v_2\}$. This reverses the arcs (u, v_1), (u, v_2), and those of the complete bipartite graph with bipartition $(\{v_1, v_2\}, X)$. Since p is even, the arc set of this bipartite graph can be decomposed into $(p-2)/2$ 4-cycles, which can be reversed one after another by Claim 8.3.1 using four inversions each. Doing so, only the arcs (u, v_1) and (u, v_2) remain reversed. ◇

Claim 8.3.3. Assume p is even. With $4(p-1)$ $(=p)$-inversions, we can reverse two non-adjacent arcs and no others.

Proof of T. o reverse two non-adjacent arcs (u_1, v_1) and (u_2, v_2), one can use Claim 8.3.2 to reverse (u_1, v_1) along with (u_1, u_2) and then (u_1, u_2) with (u_2, v_2). This can be done with $2{\cdot}2(p-1) = 4(p-1)$ $(=p)$-inversions. ◇

The construction of π satisfying (8.2) is split into four cases, according to the value of p modulo 4.

Case $p \equiv 2 \mod 4$.

In this case, we take $t = 0$, and so for every $\mathbf{u} \in \mathbb{F}_2^{\binom{n}{2}}$, $\pi(\mathbf{u}) = 0$. Concretely, (8.1) means that it is always possible to transform an n-vertex tournament into another using $(=p)$-inversions. To prove (8.2), it is enough to show that $\dim(\mathrm{Span}\{\mathbf{i}_X \mid X \subseteq [n], |X| = p\}) = \binom{n}{2}$. For every $a, b \in [n]$ with $a < b$, let $\mathbf{v}_{ab} \in \mathbb{F}_2^{\binom{n}{2}}$ be defined by, for every $i, j \in [n]$ with $i < j$,

$$(\mathbf{v}_{ab})_{\{i,j\}} = \begin{cases} 1 & \text{if } (a,b) = (i,j), \\ 0 & \text{otherwise.} \end{cases}$$

Clearly, the $\mathbf{v}_{ab}$ for $a, b \in [n]$ with $a < b$ form a family of linearly independent vectors of $\mathbb{F}_2^{\binom{n}{2}}$. We will show that they all belong to $\mathrm{Span}\{\mathbf{i}_X \mid X \subseteq [n], |X| = p\}$. Let $a, b \in [n]$ with $a < b$ and let $U \subseteq [n]$ be a $(p+2)$-set which contains a, b. We claim that

$$\mathbf{v}_{ab} = \sum_{X \subseteq U;\, |X|=p;\, a,b \in X} \mathbf{i}_X.$$

Indeed, for every $i, j \in U$ with $i < j$,

$$|\{X \subseteq U \mid |X| = p; a, b, i, j \in X\}| = \begin{cases} \binom{p+2-4}{p-4} & \text{if } i, j, a, b \text{ are distinct,} \\ \binom{p+2-3}{p-3} & \text{if } |\{i,j\} \cap \{a,b\}| = 1, \\ \binom{p+2-2}{p-2} & \text{if } \{i,j\} = \{a,b\}, \end{cases}$$

which is odd if and only if $\{i,j\} = \{a,b\}$ since $p \equiv 2 \mod 4$. This proves that $\mathbf{v}_{ab} = \sum_{X \subseteq U;\, |X|=p;\, a,b \in X} \mathbf{i}_X$, and so (8.2) holds. We deduce the following.

Corollary 8.3.8. *Let p be an integer such that $p \equiv 2 \mod 4$. Every tournament of order at least $p+2$ is $(=p)$-invertible.*

Case $p \equiv 0 \mod 4$**.**

Let $t = 1$, and let π be defined as follows. For every $\mathbf{u} \in \mathbb{F}_2^{\binom{n}{2}}$, let

$$\pi(\mathbf{u}) = \sum_{i,j \in [n], i<j} \mathbf{u}_{\{i,j\}}.$$

Less formally, π is defined so that for any T, $\pi(\mathbf{a}_T)$ is the parity of the number of backward arcs (since the operations are in $\mathbb{F}_2$).

First, since $p \equiv 0 \mod 4$, $\binom{p}{2}$ is even, and so $\pi(\mathbf{i}_X) = 0$ for every $(=p)$-set $X \subseteq [n]$. Hence, since π is clearly surjective, and by the rank theorem, to prove (8.2), it is now enough to show that $\dim(\mathrm{Span}\{\mathbf{i}_X \mid X \subseteq [n], |X| = p\}) \geq \binom{n}{2} - 1$. For every $a,b \in [n]$ with $a < b$ and $(a,b) \neq (1,2)$, let $\mathbf{w}_{ab} \in \mathbb{F}_2^n$ be the vector defined by

$$(\mathbf{w}_{ab})_{\{i,j\}} = \begin{cases} 1 & \text{if } (i,j) = (a,b) \text{ or } (i,j) = (1,2), \\ 0 & \text{otherwise,} \end{cases}$$

for every $i,j \in [n]$ with $i < j$. By Claim 8.3.2 and Claim 8.3.3, $\mathbf{w}_{ab} \in \mathrm{Span}\{\mathbf{i}_X \mid X \subseteq [n], |X| = p\}$. For every $(\alpha_{ab})_{a,b \in [n],\, a<b,\, (a,b) \neq (1,2)} \in \mathbb{F}_2^{\binom{n}{2}-1}$, if

$$\sum_{a,b \in [n], a<b, (a,b) \neq (1,2)} \alpha_{ab} \mathbf{w}_{ab} = \mathbf{0},$$

then for every $a,b \in [n], a < b, (a,b) \neq (1,2)$, the coordinate corresponding to (a,b) gives $\alpha_{ab} = 0$. Hence the vectors $(\mathbf{w}_{ab})_{a,b \in [n], a<b, (a,b) \neq (1,2)}$ are linearly independent and belong to $\mathrm{Span}\{\mathbf{i}_X \mid X \subseteq [n], |X| = p\}$. This proves $\dim(\mathrm{Span}\{\mathbf{i}_X \mid X \subseteq [n], |X| = p\}) \geq \binom{n}{2} - 1$ and so (8.2).

Corollary 8.3.9. *Let p be an integer such that $p \equiv 0 \mod 4$. Every tournament of order at least $p+2$ is $(=p)$-invertible.*

Proof. Let n be an integer such that $n \geq p+2$. It is enough to show that there are two transitive tournaments T_0, T_1 on $[n]$ such that $\pi(\mathbf{a}_{T_\epsilon}) = \epsilon$ for each $\epsilon \in \mathbb{F}_2$. We take $A(T_0) = \{ij \mid 1 \leq i < j \leq n\}$ and $A(T_1) = \{ij \mid 1 \leq i < j \leq n\} \setminus \{12\} \cup \{21\}$. □

With Lemma 8.3.13, this immediately yields the following.

Corollary 8.3.10. *Let p be an integer such that $p \equiv 0 \mod 4$. Every oriented graph of order at least $p+2$ is $(=p)$-invertible.*

Case $p \equiv 3 \mod 4$**.**

Let π be defined as follows. For every $\mathbf{u} \in \mathbb{F}_2^{\binom{n}{2}}$, let $\pi(\mathbf{u})$ be the vector of $\mathbb{F}_2^{n-1}$ defined as:

$$\pi(\mathbf{u}) = \left(\sum_{j \in [n] \setminus \{i\}} \mathbf{u}_{\{i,j\}}\right)_{i \in [n-1]}.$$

Less formally, π is such that for any tournament T, $\pi(\mathbf{a}_T)_i$ is the parity of the number of backward arcs incident to i (as tail or head). Therefore,

$$\begin{aligned}\pi(\mathbf{a}_T)_i &= |N_T^+(i) \cap [i-1]| + |N_T^-(i) \cap \{i+1, \dots, n\}| \mod 2 \\ &= |N_T^+(i) \cap [i-1]| + n - i - |N_T^+(i) \cap \{i+1, \dots, n\}| \mod 2 \\ &= n + i + d_T^+(i) \mod 2\end{aligned}$$

for every $i \in [n-1]$, which is indeed invariant, in $\mathbb{F}_2$, under $(= p)$-inversions, since p is odd. From the algebraic point of view, this means that for every $(= p)$-set $X \subseteq [n]$, for every $i \in [n-1]$, $\sum_{j \in [n] \setminus \{i\}} (\mathbf{i}_X)_{\{i,j\}} = 0$. Therefore, $\pi(\mathbf{i}_X) = \mathbf{0}$ for every $X \subseteq [n]$ with $|X| = p$. Thus, by the rank theorem, to prove (8.2), it is now enough to show that

1. π is surjective, and
2. $\dim(\mathrm{Span}\{\mathbf{i}_X \mid X \subseteq [n], |X| = p\}) \geq \binom{n}{2} - n + 1 = \binom{n-1}{2}$.

First we show that π is surjective. For every $a \in [n] \setminus \{1\}$, let $\mathbf{y}_a \in \mathbb{F}_2^{\binom{n}{2}}$ be defined by

$$(\mathbf{y}_a)_{\{i,j\}} = \begin{cases} 1 & \text{if } (i,j) = (1,a), \\ 0 & \text{otherwise,} \end{cases}$$

for every $i, j \in [n]$ with $i < j$. Then the matrix whose columns are $\pi(\mathbf{y}_a)$ for $a \in [n] \setminus \{1\}$ is, up to a permutation of the columns, the identity matrix. We conclude that π is surjective.

We now show that $\dim(\mathrm{Span}\{\mathbf{i}_X \mid X \subseteq [n], |X| = p\}) \geq \binom{n-1}{2}$. Let $a, b \in [n] \setminus \{1\}$ with $a < b$. Let $\mathbf{z}_{ab} \in \mathbb{F}_2^{\binom{n}{2}}$ be defined by

$$(\mathbf{z}_{ab})_{\{i,j\}} = \begin{cases} 1 & \text{if } (i,j) \in \{(1,2), (2,a), (a,b), (b,1)\}, \\ 0 & \text{otherwise,} \end{cases}$$

for every $i, j \in [n]$ with $i < j$. First, if $a > 2$, then $\{a, b\} \cap \{1, 2\} = \emptyset$, and so $\mathbf{z}_{ab} \in \mathrm{Span}\{\mathbf{i}_X \mid X \subseteq [n], |X| = p\}$ by Claim 8.3.1. Suppose now that $a = 2$. Let $U \subseteq [n]$ be of size $p + 2$ containing $1, 2, b$. We claim that

$$\mathbf{z}_{ab} = \sum_{X \subseteq U; |X| = p; 1,2,b \in X} \mathbf{i}_X.$$

Indeed, for every $i, j \in U$ with $i < j$,

$$|\{X \subseteq U \mid |X| = p; 1, 2, b, i, j \in X\}| = \begin{cases} \binom{p+2-5}{p-5} & \text{if } \{i,j\} \cap \{1,2,b\} = \emptyset, \\ \binom{p+2-4}{p-4} & \text{if } |\{i,j\} \cap \{1,2,b\}| = 1, \\ \binom{p+2-3}{p-3} & \text{if } \{i,j\} \subseteq \{1,2,b\}, \end{cases}$$

which is odd if and only if $\{i,j\} \subseteq \{1,2,b\}$ since $p \equiv 3 \mod 4$. It remains to show that $\mathbf{z}_{ab}$ for $a,b \in [n] \setminus \{1\}$ with $a < b$ are linearly independent. Suppose that $\alpha_{ab} \in \mathbb{F}_2$ for $a,b \in [n] \setminus \{1\}, a < b$ are such that

$$\sum_{a,b \in [n] \setminus \{1\}, a<b} \alpha_{ab} \mathbf{z}_{ab} = \mathbf{0}.$$

Then for every $a,b \in [n] \setminus \{1\}$ with $a < b$, the coordinate corresponding to $\{a,b\}$ implies that $\alpha_{ab} = 0$. This proves that the vectors $\mathbf{z}_{ab}$ for $a,b \in [n] \setminus \{1\}$ with $a < b$ are linearly independent, and concludes the proof of (8.2).

A vertex in a digraph D is *out-even* (resp. *out-odd*) if its out-degree in D is even (resp. odd).

Corollary 8.3.11. *Let p be a positive integer with $p \equiv 3 \mod 4$. For every tournament T on $n \geq p+2$ vertices, T is $(=p)$-invertible if and only if it has $\lceil n/2 \rceil$ out-even vertices.*

Proof. For every $\mathbf{u} \in \mathbb{F}_2^{\binom{n}{2}}$, let $\pi'(\mathbf{u}) \in \mathbb{F}_2^n$ be defined by

$$\pi'(\mathbf{u})_i = \pi(\mathbf{u})_i + n + i \mod 2,$$

for every $i \in [n-1]$, and

$$\pi'(\mathbf{u})_n = \binom{n}{2} + \sum_{i \in [n-1]} \pi'(u)_i \mod 2.$$

We claim that for every tournament T on $[n]$, for every $i \in [n]$, $\pi'(\mathbf{a}_T)_i = d_T^+(i) \mod 2$. Indeed, letting $d_L^+(i)$ (resp. $d_R^+(i)$) be the number of out-neighbours of i with index $j < i$ (resp. $j > i$), and $d_R^-(i)$ the number of in-neighbours with $j > i$, we obtain the following equalities modulo 2:

$$d^+(i) = d_L^+ + d_R^+ + 2d_R^-(i) = \pi_i(u) + d_R(i) = \pi_i(u) + n - i = \pi_i(u) + n + i$$

Therefore, $\pi'(\mathbf{a}_T)_i = 0$ if vertex i is out-even and $\pi'(\mathbf{a}_T)_i = 1$ if vertex i is out-odd. (For $i = n$, it follows from the equalities for smaller values of i and the fact that $\sum_{i \in [n]} d_T^+(i) = \binom{n}{2}$.) Moreover, for every $\mathbf{u}, \mathbf{u}' \in \mathbb{F}_2^{\binom{n}{2}}$, $\pi'(\mathbf{u}) = \pi'(\mathbf{u}')$ if and only if $\pi(\mathbf{u}) = \pi(\mathbf{u}')$. In turn, it remains to show that for every $\mathbf{v} \in \mathbb{F}_2^{n-1}$, there exists a transitive tournament T on $[n]$ such that $\pi'(\mathbf{a}_T) = \mathbf{v}$ if and only if the number of zeros in $\mathbf{v}$ is $\lceil n/2 \rceil$.

First, for every transitive tournament T, the number of vertices of even out-degree is $\lceil n/2 \rceil$, and so the number of zeros in $\pi(\mathbf{a}_T)$ is $\lceil n/2 \rceil$. Conversely, for every $\mathbf{v} \in \mathbb{F}_2^{n-1}$ with $\lceil n/2 \rceil$ zeros, let σ be a permutation of $[n]$ such that for every $i \in [n]$, $\mathbf{v}_{\sigma(i)} = 1$ if and only if $n + i$ is odd. Then, $T = ([n], \{\sigma(i)\sigma(j) \mid 1 \leq i < j \leq n\})$ is a transitive tournament with $\pi'(\mathbf{a}_T) = \mathbf{v}$. This proves the corollary. □

Case $p \equiv 1 \mod 4$.

We define π as follows. For every $\mathbf{u} \in \mathbb{F}_2^{\binom{n}{2}}$, let

$$\pi(\mathbf{u}) = \left(\left(\textstyle\sum_{j \in [n] \setminus \{i\}} \mathbf{u}_{\{i,j\}} \right)_{i \in [n-1]}, \textstyle\sum_{i,j \in [n], i<j} \mathbf{u}_{\{i,j\}} \right).$$

Since $p \equiv 1 \mod 4$, $\binom{p}{2}$ is even and so $\sum_{i,j\in[n],i<j}(\mathbf{i}_X)_{\{i,j\}} = 0$ for every $X \subseteq [n]$ with $|X| = p$. Moreover, since p is odd, for every $i \in [n-1]$, $\sum_{j\in[n]\setminus\{i\}}(\mathbf{i}_X)_{\{i,j\}} = 0$. Therefore, $\pi(\mathbf{i}_X) = \mathbf{0}$ for every $(= p)$-set $X \subseteq [n]$. Thus, to prove (8.2), by the rank theorem, it is now enough to show that

1. π is surjective, and
2. $\dim(\mathrm{Span}\{\mathbf{i}_X \mid X \subseteq [n], |X| = p\}) \geq \binom{n}{2} - n$.

First we show that π is surjective. For every $a \in [n-1]$, let $\mathbf{t}_a \in \mathbb{F}_2^{\binom{n}{2}}$ be defined by

$$(\mathbf{t}_a)_{\{i,j\}} = \begin{cases} 1 & \text{if } (i,j) = (a,n), \\ 0 & \text{otherwise,} \end{cases}$$

for every $i, j \in [n]$ with $i < j$. Moreover, let $\mathbf{t}_n \in \mathbb{F}_2^{\binom{n}{2}}$ be defined by

$$(\mathbf{t}_n)_{\{i,j\}} = \begin{cases} 1 & \text{if } (i,j) \in \{(1,2),(1,3),(2,3)\} \\ 0 & \text{otherwise,} \end{cases}$$

for every $i, j \in [n]$ with $i < j$. The matrix whose columns are $(\pi(\mathbf{t}_a))_{a\in[n]}$ is, up to the ordering of the columns,

$$\begin{pmatrix} 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & & 0 & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & \dots & 1 & 0 \\ 1 & 1 & \dots & 1 & 1 \end{pmatrix}$$

and so π is surjective.

We now show that $\dim(\mathrm{Span}\{\mathbf{i}_X \mid X \subseteq [n], |X| = p\}) \geq \binom{n}{2} - n$. For every $a, b \in [n] \setminus \{1,2\}$ with $a < b$, let $\mathbf{s}_{ab} \in \mathbb{F}_2^{\binom{n}{2}}$ be defined by

$$(\mathbf{s}_{ab})_{\{i,j\}} = \begin{cases} 1 & \text{if } (i,j) \in \{(1,2),(2,b),(a,b),(1,b)\}, \\ 0 & \text{otherwise,} \end{cases}$$

for every $i, j \in [n]$ with $i < j$. By Claim 8.3.1, $\mathbf{s}_{ab} \in \mathrm{Span}\{\mathbf{i}_X \mid X \subseteq [n], |X| = p\}$. Moreover, for every $a \in \{3, \dots, n-1\}$, let $\mathbf{s}_a \in \mathbb{F}_2^{\binom{n}{2}}$ be defined by

$$(\mathbf{s}_a)_{\{i,j\}} = \begin{cases} 1 & \text{if } (i,j) \in \{(1,2),(2,a),(a,a+1),(1,a+1)\}, \\ 0 & \text{otherwise,} \end{cases}$$

for every $i, j \in [n]$ with $i < j$. Again, by Claim 8.3.1, $\mathbf{s}_a \in \mathrm{Span}\{\mathbf{i}_X \mid X \subseteq [n], |X| = p\}$. We claim that the vectors $(\mathbf{s}_{ab})_{a,b\in[n]\setminus\{1,2\},a<b}, (\mathbf{s}_a)_{a\in\{3,\dots,n-1\}}$ are linearly independent. Suppose for contradiction that there exists $\alpha_{ab} \in \mathbb{F}_2$ for $a, b \in [n]\setminus\{1,2\}$, $a < b$, and $\alpha_a \in \mathbb{F}_2$ for $a \in \{3, \dots, n-1\}$ not all zero such that

$$\sum_{a,b\in[n]\setminus\{1,2\},a<b} \alpha_{ab}\mathbf{s}_{ab} + \sum_{a\in\{3,\dots,n-1\}} \alpha_a\mathbf{s}_a = \mathbf{0}.$$

First assume that $\alpha_a = 0$ for every $a \in \{3, \dots, n-1\}$. Then for every $a, b \in [n] \setminus \{1, 2\}$ with $a < b$, the coordinate corresponding to $\{a, b\}$ implies that $\alpha_{ab} = 0$. This is a contradiction.

Now suppose that there exists $a_0 \in \{3, \dots, n-1\}$ such that $\alpha_{a_0} = 1$. Take such an a_0 maximum. The coordinate corresponding to $\{a_0, a_0+1\}$ implies that $\alpha_{(a_0, a_0+1)} = 1$. But then, the coordinate corresponding to $\{1, a_0+1\}$ implies that there exists $b \in [n] \setminus \{1, 2\}$ with $b > a_0 + 1$ such that $\alpha_{(a_0+1, b)} = 1$. As a consequence, the coordinate corresponding to $\{a_0+1, b\}$ implies that $\alpha_{b-1} = 1$, contradicting the maximality of a_0.

Therefore, the vectors $(\mathbf{s}_{ab})_{a,b \in [n] \setminus \{1,2\}, a<b}, (\mathbf{s}_a)_{a \in \{3,\dots,n-1\}}$ are linearly independent and so

$$\dim(\mathrm{Span}\{\mathbf{i}_X \mid X \subseteq [n], |X| = p\}) \geq \binom{n-2}{2} + n - 3 = \binom{n}{2} - n.$$

This proves (8.2).

Corollary 8.3.12. *Let p be a positive integer with $p \equiv 1 \mod 4$. For every tournament T on $n \geq p+2$ vertices, T is $(=p)$-invertible if and only if it has $\lceil n/2 \rceil$ out-even vertices.*

Proof. For every $\mathbf{u} \in \mathbb{F}_2^{\binom{n}{2}}$, let $\pi'(\mathbf{u}) \in \mathbb{F}_2^{n+1}$ be defined by

$$\pi'(\mathbf{u})_i = \pi(\mathbf{u})_i + n + i \mod 2,$$

for every $i \in [n-1]$,

$$\pi'(\mathbf{u})_n = \binom{n}{2} + \sum_{i \in [n-1]} \pi'(u)_i \mod 2,$$

and

$$\pi'(\mathbf{u})_{n+1} = \pi(\mathbf{u})_n.$$

Observe that for every tournament T on the vertex set $[n]$, for every $i \in [n]$, $\pi'(\mathbf{a}_T)_i = d_T^+(i) \mod 2$, and $\pi'(\mathbf{a}_T)_{n+1}$ is the parity of the number of backward arcs. Moreover, for every $\mathbf{u}, \mathbf{u}' \in \mathbb{F}_2^{\binom{n}{2}}$, $\pi'(\mathbf{u}) = \pi'(\mathbf{u}')$ if and only if $\pi(\mathbf{u}) = \pi(\mathbf{u}')$. Now, it remains to show that for every $\mathbf{v} \in \mathbb{F}_2^{n-1}$, there exists a transitive tournament T on $[n]$ such that $\pi'(\mathbf{a}_T) = \mathbf{v}$ if and only if the number of zeros in $\{\mathbf{v}_i \mid i \in [n]\}$ is $\lceil n/2 \rceil$.

First, for every transitive tournament T on $[n]$, the number of out-even vertices is $\lceil n/2 \rceil$, and so the number of zeros in $\pi'(\mathbf{a}_T)$ is $\lceil n/2 \rceil$. Reciprocally, for every $\mathbf{v} \in \mathbb{F}_2^{n+1}$ with exactly $\lceil n/2 \rceil$ zeros in $(\mathbf{v}_1, \dots, \mathbf{v}_n)$, let σ be a permutation of $[n]$ such that for every $i \in [n]$, $\mathbf{v}_{\sigma(i)} = 1$ if and only if $n + i$ is odd. By possibly exchanging the position of 1 and 3 in σ, we can assume that $|\{ij \mid 1 \leq i < j \leq n, \sigma(i) > \sigma(j)\}| \mod 2 = \mathbf{v}_{n+1}$. Then, $T = ([n], \{\sigma(i)\sigma(j) \mid i < j\})$ is a transitive tournament with $\pi'(\mathbf{a}_T) = \mathbf{v}$. This proves the corollary. □

8.3.3 Polynomial-time algorithm

In this subsection, we shall use the results of the previous subsection to prove the following theorem.

Theorem 8.1.2. *There is a polynomial-time algorithm that, given a positive integer p and an oriented graph D of order distinct from $p+1$, decides whether D is $(=p)$-invertible.*

We need the following lemma that will allow us to reduce to tournaments.

Lemma 8.3.13. *An oriented graph D is $(=p)$-invertible if and only if there is an $(=p)$-invertible tournament T having D as spanning subdigraph.*

Proof. If a tournament T admits D as a spanning subdigraph, any decycling $(=p)$-family of T is also a decycling $(=p)$-family of D. Conversely, assume D admits a decycling $(=p)$-family $\mathcal{X}$. Let $D' = \mathrm{Inv}(D; \mathcal{X})$. Since it is acyclic, D' admits an acyclic ordering σ. Let T' be the transitive tournament on $V(D)$ with acyclic ordering σ and let $T = \mathrm{Inv}(T'; \mathcal{X})$. Observe now that D is a spanning subgraph of T. Moreover $\mathrm{Inv}(T; \mathcal{X}) = T'$, so $\mathcal{X}$ a decycling $(=p)$-family of T. $\square$

With Corollaries 8.3.8 and 8.3.9, this immediately yields the following for even p.

Corollary 8.3.14. *Let p be an even integer. Every oriented graph of order at least $p+2$ is $(=p)$-invertible.*

When p is odd, we need the following lemma.

Lemma 8.3.15. *There is a polynomial-time algorithm that, given a graph G with a partition (A_1, A_2) of $V(G)$ and an integer s, either returns an orientation $\overrightarrow{G}$ of G such that the set $S_{\overrightarrow{G}} := \{v \in A_1 \mid v \text{ is out-odd in } \overrightarrow{G}\} \cup \{v \in A_2 \mid v \text{ is out-even in } \overrightarrow{G}\}$ has size s, or returns "No" if no such orientation exists.*

Proof. Let G be a graph, (A_1, A_2) a partition of $V(G)$ and s an integer. Before describing our procedure, we observe the following:

(i) For any orientation $\overrightarrow{G}$ of G the parity of $|S_{\overrightarrow{G}}|$ is the same, since reversing an arc does not change it.

(ii) Reversing an arc with one vertex in $S_{\overrightarrow{G}}$ and the other not in $S_{\overrightarrow{G}}$ leaves the size of $S_{\overrightarrow{G}}$ unchanged.

(iii) Reversing any arc with both ends inside (resp. both out-side) of $S_{\overrightarrow{G}}$ decreases (resp. increases) the size of $S_{\overrightarrow{G}}$ by 2.

(iv) In the same way reversing a path between two vertices w, z outside (resp. inside) $S_{\overrightarrow{G}}$ with all internal vertices inside (resp.outside) $S_{\overrightarrow{G}}$ adds w, z to $S_{\overrightarrow{G}}$ (resp. removes w, z from $S_{\overrightarrow{G}}$).

With this in hand, we start with an initial arbitrary orientation $\overrightarrow{G}$. If $|S_{\overrightarrow{G}}| \not\equiv s \mod 2$, then we return "No", according to (i). Otherwise $|S_{\overrightarrow{G}}| \equiv s \mod 2$, and we distinguish two cases depending on whether $|S_{\overrightarrow{G}}| \leq s$ or not.

If $|S_{\overrightarrow{G}}| \leq s$, we apply the following steps.

1. If $|S_{\overrightarrow{G}}| = s$, then return $\overrightarrow{G}$.
2. If every connected component of G contains at most one vertex not in $S_{\overrightarrow{G}}$, then return "No", since (ii) and (iii) yield that $S_{\overrightarrow{G}}$ is already of maximum size.
3. Let C be a connected component of G which contains at least two vertices $w, z \notin S_{\overrightarrow{G}}$. Take a minimum (undirected) path P from w to z, and note that all internal vertices of P must be in $S_{\overrightarrow{G}}$. Replace $\overrightarrow{G}$ by the orientation obtained from it by reversing the arcs of P and go to step 1.

If $|S_{\overrightarrow{G}}| \geq s$, we apply the following similar steps.

1. If $|S_{\overrightarrow{G}}| = s$, then return $\overrightarrow{G}$.
2. If every connected component of G contains at most one vertex in $S_{\overrightarrow{G}}$, then return "No", since since (ii) and (iii) yield that $S_{\overrightarrow{G}}$ is already of minimum size.
3. Switch the roles of $S_{\overrightarrow{G}}$ and its complement and proceed as in the case when $|S_{\overrightarrow{G}}| \leq s$.

Now, when $|S_{\overrightarrow{G}}| \leq s$ (resp. $|S_{\overrightarrow{G}}| \geq s$), if we never output "No", each step decreases (resp. increases) $|S_{\overrightarrow{G}}|$ by two, according to (iii) and (iv), so the process must end when $|S_{\overrightarrow{G}}| = s$. □

We are now ready to prove Theorem 8.1.2.

Proof of Theorem 8.1.2. Let D be an oriented graph on n vertices and p a positive integer. If $n \leq p - 1$, then there is no $(= p)$-subsets of $[n]$, so D is $(= p)$-invertible if and only if it is acyclic.

From now on, we may assume $n \geq p + 2$.

If p is even, then by Corollary 8.3.14, D is $(= p)$-invertible. Now assume p odd.

Let G be the complement of the underlying graph of D, that is, such that $V(G) = V(D)$ and $uv \in E(G)$ if and only if there is no arc between u and v in D. Let A_1 (resp. A_2) be the set of out-odd (resp out-even) vertices in D. By Lemma 8.3.13, Corollary 8.3.11, and Corollary 8.3.12, D is $(= p)$-invertible if and only if there is a tournament T with D as a spanning subdigraph and $\lceil n/2 \rceil$ out-even vertices. Observe that for any tournament T containing D as a subgraph, $\overrightarrow{G} = (V(D), A(T) \setminus A(D))$ is an orientation of G, and that the set of out-even vertices in T is $S_{\overrightarrow{G}} = \{v \in A_1 \mid v \text{ is out-odd in } \overrightarrow{G}\} \cup \{v \in A_2 \mid v \text{ is out-even in } \overrightarrow{G}\}$. Therefore D is $(= p)$-invertible if and only if there is an orientation $\overrightarrow{G}$ of G such that $|S_{\overrightarrow{G}}| = \lceil n/2 \rceil$. This can be checked in polynomial time by Lemma 8.3.15. □

8.4 Bounds on $(= p)$-inversion numbers

Once we know whether an oriented graph D is $(= p)$-invertible or not, two different questions arise. If D is $(= p)$-invertible, what is the $(= p)$-inversion number of D? And if D is not $(= p)$-invertible, how close to an acyclic digraph can it be made? Of course, for the latter, we need a measure of closeness to acyclicity. A natural one is the feedback-arc-set number.

Proposition 8.4.1. *Let p be an odd integer such that $p \geq 3$. Every oriented graph of order $n \geq p + 2$ can be transformed using $(= p)$-inversions in a digraph with feedback-arc-set number at most $\frac{1}{2}\lceil n/2 \rceil$.*

Proof. Let D be an oriented graph of order n. Let T a tournament of order n such that D is a subdigraph of T. Since an arc-reversal changes the parity of the out-degree of exactly two vertices, one could reverse a set R of at most $\frac{1}{2}\lceil n/2 \rceil$ arcs in T to obtain a tournament T' having exactly $\frac{1}{2}\lceil n/2 \rceil$ out-even vertices. Now, by Corollaries 8.3.11 and 8.3.12, T' is $(= p)$-invertible, so there is family $\mathcal{X}$ of $(= p)$-sets such that $\text{Inv}(T'; \mathcal{X})$ is acyclic. Then R is a feedback arc-set of $\text{Inv}(T; \mathcal{X})$, and $R \cap A(D)$ a feedbak arc-set of $\text{Inv}(D; \mathcal{X})$. □

The above proposition is tight. Indeed, consider a k-diregular tournament T for k odd. Its number of vertices is $n = 2k + 1$. Let T' be a tournament obtained from T using $(= p)$-inversions for some odd p. Since T has no out-even vertices, so does T'. Now a transitive tournament on n vertices has $\lceil n/2 \rceil$ out-even vertices, and an arc-reversal changes the parity of the out-degree of two vertices. Therefore, $\text{fas}(T') \geq \frac{1}{2}\lceil n/2 \rceil$.

We shall now consider the first question.

8.4.1 Bounds on the $(= p)$-inversion number.

First bound in terms of the size

The algebraic point of view used in Subsection 8.3.2, in particular Property (8.2), has an immediate consequence on number of $(= p)$-inversions to go from an oriented graph to another (when it is possible):

Theorem 8.4.2. *For every orientations $\overrightarrow{G}_1, \overrightarrow{G}_2$ of the same graph G, if there exists sets $X_1, \ldots, X_\ell \subseteq V(G)$, all of size p, such that $\overrightarrow{G}_1 = \text{Inv}(\overrightarrow{G}_2; X_1, \ldots, X_\ell)$, then there exists sets $X'_1, \ldots, X'_{\ell'} \subseteq V(G)$, all of size p, such that $\overrightarrow{G}_1 = \text{Inv}(\overrightarrow{G}_2; X'_1, \ldots, X'_{\ell'})$ and*

$$\ell' \leq |E(G)|.$$

Proof. Let $\mathbf{a} \in \mathbb{F}_2^{E(G)}$ be defined by

$$\mathbf{a}_e = \begin{cases} 0 & \text{if } e \text{ has the same orientation in } \overrightarrow{G}_1 \text{ and } \overrightarrow{G}_2 \\ 1 & \text{otherwise.} \end{cases}$$

for every $e \in E(G)$. Moreover, for every $X \subseteq V(G)$, let $\mathbf{i}_X \in \mathbb{F}_2^{E(G)}$ be defined by

$$(\mathbf{i}_X)_{uv} = \begin{cases} 1 & \text{if } u, v \in X \\ 0 & \text{otherwise.} \end{cases}$$

for every $uv \in E(G)$. Then we have, for every $X_1, \ldots, X_\ell \subseteq V(G)$,

$$\mathbf{a} = \sum_{j \in [\ell]} \mathbf{i}_{X_j} \Leftrightarrow \overrightarrow{G}_1 = \text{Inv}(\overrightarrow{G}_2; X_1, \ldots, X_\ell).$$

Let $X_1, \ldots, X_\ell \subseteq V(G)$ all of size p such that $\overrightarrow{G}_1 = \text{Inv}(\overrightarrow{G}_2; X_1, \ldots, X_\ell)$ and ℓ is minimum. We want to show that $\ell \leq |E(G)|$. Suppose for contradiction that $\ell > |E(G)|$. Since $\ell > |E(G)| = \dim \mathbb{F}_2^{E(G)}$, the family $(\mathbf{i}_{X_j})_j$ is dependent over $\mathbb{F}_2^{E(G)}$, so there exists $J \subseteq [\ell]$ nonempty such that $\sum_{j \in J} \mathbf{i}_{X_j} = \mathbf{0}$. It follows that

$$\mathbf{a} = \sum_{j \in [\ell] \setminus J} \mathbf{i}_{X_j}.$$

and so $\overrightarrow{G}_1 = \text{Inv}(\overrightarrow{G}_2; (X_j)_{j \in [\ell] \setminus J})$, contradicting the minimality of ℓ. □

Corollary 8.4.3. *Let D be an oriented graph. If D is $(= p)$-invertible, then* $\text{inv}^{=p}(D) \leq |A(D)|$.

First bound in terms of the feedback-arc-set number and better bound in terms of the size

Theorem 8.4.4. *Let $p \geq 4$ be an even integer and let D be an oriented graph with at least $p+2$ vertices. Then,*

$$\mathrm{inv}^{=p}(D) \leq 2(p-1)(\mathrm{fas}(D)+1).$$

Proof. Let D_1 be the acyclic oriented graph obtained by reversing every arc of a minimum feedback arc set of D and let D_2 be an acyclic oriented graph obtained from D_1 by reversing exactly one arc. Then D disagrees with one of D_1, D_2 in a set E' of arcs such that $|E'|$ is even and $|E'| \leq \mathrm{fas}(D)+1$.

By Claims 8.3.2 and 8.3.3, one can reverse the arcs of $|E'|$ two by two using at most $4(p-1)\frac{|E'|}{2}$ $(=p)$-inversions. Thus $\mathrm{inv}^{=p}(D) \leq 2(p-1)(\mathrm{fas}(D)+1)$. □

Corollary 8.4.5. *Let $p \geq 4$ be an even integer, and let D be an oriented graph with at least $p+2$ vertices and at least one directed cycle. Then,*

$$\mathrm{inv}^{=p}(D) \leq \mathrm{inv}^{\leq p}(D)4(p-1)\binom{p}{2}.$$

Proof. Note that $\mathrm{inv}^{\leq p}(D) \geq \frac{\mathrm{fas}(D)}{\binom{p}{2}}$, and using this we argue as follows.

$$\begin{aligned}
\mathrm{inv}^{\leq p}(D)4(p-1)\binom{p}{2} &\geq 4(p-1)\binom{p}{2}\frac{\mathrm{fas}(D)}{\binom{p}{2}} \\
&\geq 4(p-1)\,\mathrm{fas}(D) \\
&\geq 4(p-1)\frac{\mathrm{fas}(D)+1}{2} \\
&\geq \mathrm{inv}^{=p}(D) && \text{by Theorem 8.4.4.}
\end{aligned}$$

□

Similarly, one can ask whether for odd p there exists a function f_p such that $\mathrm{inv}^{=p}(D) \leq f_p(\mathrm{inv}^{\leq p}(D))$ for all $(=p)$-invertible oriented graph D. It turns out that the answer for this question is negative, even when D is a tournament and $p=3$.

Proposition 8.4.6. *Let p be an odd positive integer. There is no function f_p such that* $\mathrm{inv}^{=p}(T) \leq f_p(\mathrm{inv}^{\leq p}(T))$ *for all $(=p)$-invertible tournament T.*

Proof. Let TT_n be a transitive tournament on n vertices. Let $v_1,\dots,v_n$ be the unique acyclic ordering of TT_n. We denote by T_n be the tournament obtained by reversing the arc v_1v_n (so $\mathrm{inv}^{\leq p}(T_n)=1$). Note that, for even n, T_n has $n/2$ out-even vertices, and by Corollary 8.3.11, T_n is $(=p)$-invertible.

Towards a contradiction, suppose that there is a function f_p such that $\mathrm{inv}^{=p}(T_n) \leq f_p(\mathrm{inv}^{\leq p}(T_n)) = f_p(1)$ for all n. Then consider a sufficiently large even n, greater than $p+2$, such that $\frac{n-3}{p} \geq f_p(1)$. Let $\mathcal{X}$ be a decycling $(=p)$-family of T_n of cardinality at most $\frac{n-3}{p}$. Since at most $n-3$ vertices can be involved in $\mathcal{X}$, there must exist a vertex v_i for $2 \leq i \leq n-1$ that does not belong to any set of $\mathcal{X}$. Thus $\mathcal{X}$ is also a decycling $(=p)$-family of $T_n - v_i = T_{n-1}$, and so T_{n-1} is $(=p)$-invertible. But, since $(n-1)$ is odd, T_{n-1} has less than $\lceil n-1/2 \rceil$ out-even vertices, a contradiction to Corollary 8.3.11. □

Lemma 8.4.7. *Let p be an integer greater than 1, and let D be an oriented graph of order n. In at most $|A(D)|/(p-1)$ $(=p)$-inversions, we transform D into an oriented graph D' with* $\text{fas}(D') \le (p-2)n - \frac{3}{4}p^2 + \frac{7p}{4}$.

Proof. Let $(v_1, \dots, v_n)$ be an ordering of the vertices of D. Let $D = D_1$, and for $k = 1$ to $n-p$ we do the following.

> Let B_k be the set of backward arcs with head v_k in D_k. Let $u_1, \dots, u_t$ be the tails of the arcs in B_k. (Note that $\{u_1, \dots, u_t\} \subseteq \{v_{k+1}, \dots, v_n\}$.) Set $s_k = \lfloor t/(p-1) \rfloor$, and for $i = 1$ to s_k, let $X_i = \{v_k\} \cup \{u_j \mid (i-1)(p-1)+1 \le j \le i(p-1)\}$. We then invert all the X_i (which are all p-sets) to obtain D_{k+1}.

Observe that for each $k \in [n-p]$, at step k, we reversed all backward arcs with head v_k except possibly $t - s_k(p-1) \le p-2$. Moreover v_k is in no inversions in further steps. So in D', there are at most $(p-2)(n-p)$ backward arcs with head in $\{v_1, \dots, v_{n-p}\}$.

Now consider a median order $(w_{n-p+1}, \dots, w_n)$ of $D_{n-p+1}[\{v_{n-p+1}, \dots, v_n\}]$. It has at most $\frac{1}{2}\binom{p}{2}$ backward arcs.

Thus, considering the ordering $(v_1, \dots, v_{n-p}, w_{n-p+1}, \dots, w_n)$, we get

$$\text{fas}(D') \le (p-2)(n-p) + \frac{1}{2}\binom{p}{2} = (p-2)n - \frac{3}{4}p^2 + \frac{7p}{4}$$

The number of inversions made to go from D to D' is $\sum_{k=1}^{n-p} s_k \le 1 + \sum_{k=1}^{n-p} |B_k|/(p-1)$. Now each arc in B_k is incident to v_k and a vertex of $\{v_{k+1}, \dots, v_n\}$, so $\sum_{k=1}^{n-p} |B_k| \le |A(D)|$. So the number of inversions is at most $|A(D)|/(p-1)$. □

Lemma 8.4.7 and Theorem 8.4.4 immediately imply the following.

Corollary 8.4.8. *Let $p \ge 4$ be an even integer and let D be an oriented graph of order $n \ge p+2$. Then,*

$$\text{inv}^{=p}(D) \le \frac{|A(D)|}{p-1} + 2(p-1)\left((p-2)n - \frac{3}{4}p^2 + \frac{7p}{4}\right)$$

A natural question is whether such a corollary also holds when p odd.

Problem 8.4.9. *Let $p \ge 3$ be an odd integer. Does there exist a constant C_p such that* $\text{inv}^{=p}(D) \le \frac{|A(D)|}{p-1} + C_p \cdot n$ *for every invertible oriented graph D of order $n \ge p+2$?*

Better bounds for dense digraphs

The following proposition is easy and well known. Its proof is left to the reader.

Proposition 8.4.10. *A graph of order n with no even cycles has at most $\lfloor \frac{3(n-1)}{2} \rfloor$ edges.*

Proposition 8.4.11. *Let $p \ge 3$ and $\ell \ge 2$ be two integers, and let D be an oriented graph of order at least $p+2$. With 4ℓ $(=p)$-inversions, we can reverse the arcs of a 2ℓ-cycle and none other.*

Proof. We may assume without loss of generality that D is a tournament, up to completing it arbitrarily to obtain one. Indeed, if the result holds for such a tournament, then applying the same inversions over D also produces the desired sol. Now, let $C = (x_1, \dots, x_{2\ell}, x_1)$ be a (initial) 2ℓ-cycle of D. Consider the 4-cycles $C_i = (x_i, x_{i+1}, x_{i+\ell+1}, x_{i+\ell}, x_i)$ (with indices modulo 2ℓ) for $i \in [\ell]$. By Claim 8.3.1, one can reverse the arcs of each C_i and no other in four $(= p)$-inversions. Doing so for each C_i, every chord of type $x_{i+\ell}x_i$ is reversed twice and each arc of C is reversed only once. Thus, those 4ℓ inversions reverse the arcs of C and no other arcs. □

Theorem 8.4.12. *Let p be an even integer greater than 2, $k \geq 1$ an integer and D an oriented graph of order $n \geq p + 2$. Then* $\mathrm{inv}^{=p}(D) \leq 2\,\mathrm{fas}(D) + 2pn - \frac{3}{2}p - 5n + \frac{13}{2}$.

Proof. Let D_1 be the acyclic oriented graph obtained by reversing every arc of a minimum feedback arc set of D and let D_2 be an acyclic oriented graph obtained from D_1 by reversing exactly one arc. Then, there is some i such that D disagrees with D_i on an even number of arcs, call them F', and we have $|F'| \leq \mathrm{fas}(D) + 1$.

Let F'_1 be a maximal subset of F' which is the union of arc-disjoint even cycles. By Proposition 8.4.10, $|F' \setminus F'_1| \leq \lfloor \frac{3(n-1)}{2} \rfloor$. Let P_2 be a maximum set of disjoint pairs of adjacent arcs in $F' \setminus F'_1$, let F'_2 be the set of arcs in pairs of P_2, and let $F'_3 = F' \setminus (F'_1 \cup F'_2)$. By our choice of P_2, the set F'_3 is a matching so $|F'_3| \leq n/2$. Moreover, $|F'_3|$ is even because F' was even and F'_1 and F'_2 are also even by definition. We first reverse the arcs of F'_1 using 4ℓ $(= p)$-inversions per 2ℓ-cycle by Proposition 8.4.11, so in average two $(= p)$-inversions per arc. We then reverse the arcs of F'_2 using $2(p-1)$ $(= p)$-inversions per pair of adjacent arcs by Claim 8.3.2, so in average $p - 1$ $(= p)$-inversions per arc. Finally, we reverse the arcs of F'_3 using $4(p-1)$ $(= p)$-inversions per pair of non-adjacent arcs by Claim 8.3.3, so in average $(2p-2)$ $(= p)$-inversions per arc. Therefore, D_1 can be transformed in the acyclic digraph D_2 (and thus D into the acyclic digraph D^*) using $2|F'_1| + (p-1)|F'_2| + (2p-2)|F'_3|$ $(= p)$-inversions. Since $|F'_2 \cup F'_3| \leq \frac{3(n-1)}{2}$, and $|F'_3| \leq n/2$, we get

$$\begin{aligned}
\mathrm{inv}^{=p}(D) &\leq 2|F'_1| + (p-1)|F'_2| + (2p-2)|F'_3| \\
&\leq 2|F'| + (p-3)|F'_2| + (2p-4)|F'_3| \\
&\leq 2\,\mathrm{fas}(D) + 2 + (p-3)|F'_2 \cup F'_2| + (p-1)|F'_3| \\
&\leq 2\,\mathrm{fas}(D) + 2 + (p-3)\frac{3(n-1)}{2} + (p-1)n/2 \\
&\leq 2\,\mathrm{fas}(D) + 2pn - \frac{3}{2}p - 5n + \frac{13}{2}
\end{aligned}$$

□

A natural question is to ask for statement similar to Theorem 8.4.12 for odd p.

Problem 8.4.13. *Let $p \geq 3$ be an odd integer, and $k \geq 1$ an integer. Do there exist constants C, C_p such that* $\mathrm{inv}^{=p}(D) \leq C \cdot \mathrm{fas}(D) + C_p \cdot n$ *for every invertible oriented graph of order $n \geq p+2$?*

Problem 8.4.14. *For p even, $n \geq p + 2$, determine*

$$\begin{aligned}
M^{=p}(n) &= \max\{\mathrm{inv}^{=p}(D) \mid D \text{ digraph of order } n\} \\
&= \max\{\mathrm{inv}^{=p}(T) \mid T \text{ tournament of order } n\}.
\end{aligned}$$

Better bounds for very dense digraphs

Theorem 8.4.15. *Let p be an integer with $p \geq 2$. There exist constants α_p and ϵ_p with $\epsilon_p > 0$ such that the following holds. Let n be a positive integer, and let $E \subseteq \binom{[n]}{2}$. There exists an integer*

ℓ with $\ell \leq \frac{|E|}{\lceil p/2 \rceil \cdot \lfloor p/2 \rfloor}$ and $(=p)$-sets $X_1, \ldots, X_\ell \subseteq [n]$ such that

$$\left| E \Delta \binom{X_1}{2} \Delta \ldots \Delta \binom{X_\ell}{2} \right| \leq \alpha_p \cdot n^{2-\epsilon_p}.$$

Intuitively, this means that after $\frac{|E|}{\lceil p/2 \rceil \cdot \lfloor p/2 \rfloor}$ inversions, we have reversed exactly the edges in E, up to $\mathcal{O}(n^{2-\epsilon_p})$ errors. The quantity $\frac{|E|}{\lceil p/2 \rceil \cdot \lfloor p/2 \rfloor}$ comes from the following observation: in a bipartite graph G, $|E(G) \cap \binom{X}{2}| \leq \lceil p/2 \rceil \cdot \lfloor p/2 \rfloor$ for every $X \subseteq V(G)$ of size at most p. Hence, $\frac{|E|}{\lceil p/2 \rceil \cdot \lfloor p/2 \rfloor}$ is a lower bound, which is not too far from the optimum by Theorem 8.4.15. We will use the following classical theorem by Kővári, Sós, and Turán [KST54].

Theorem 8.4.16 (Kővári, Sós, and Turán '54)**.** *Let s, t be positive integers with $t \leq s$, and let G be a graph on n vertices. If $|E(G)| > 4s^{\frac{1}{t}} n^{2-\frac{1}{t}}$, then $K_{s,t}$ is a subgraph of G.*

Proof of Theorem 8.4.15. Note that it is enough to prove the result for large enough values of n. Let

$$s = \left\lfloor n^{\frac{1}{2t^2}} \right\rfloor, \quad t = \left\lfloor \frac{p}{2} \right\rfloor, \quad N = \left\lfloor n^{2-\frac{1}{3t}} \right\rfloor, \quad C = \left\lfloor n^{1-\frac{1}{2t}} \right\rfloor,$$

and let n_0 be a positive integer such that

$$n^{1-\frac{1}{2t}} \left(\ln n - 2^{-2\lceil \frac{p}{2} \rceil - 1} n^{\frac{1}{6t}} \right) < -\ln(2n)$$

for every $n \geq n_0$.

Let n be an integer with $n \geq n_0$, and let $\mathcal{Y}$ be a random family of subsets of $V(G)$ each of size $\lceil \frac{p}{2} \rceil$, such that the event $X \in \mathcal{Y}$ for $X \in \binom{V(G)}{\lceil \frac{p}{2} \rceil}$ are independent and with the same probability $N \cdot \binom{n}{\lceil \frac{p}{2} \rceil}^{-1}$.

Let $G = ([n], E)$. We will build a sequence $H_1, \ldots, H_r$ of pairwise edge-disjoint copies of $K_{s,t}$ in G as follows.

Let $i \geq 1$. Assume that $H_1, \ldots, H_{i-1}$ is already constructed. If $G \setminus \bigcup_{j \in [i-1]} E(H_j)$ does not contain $K_{t,s+C}$ as a subgraph, then the sequence is finished. Otherwise, let H_i^0 be a copy of $K_{t,s+C}$ in $G \setminus \bigcup_{j \in [i-1]} E(H_j)$ with bipartition (A_i, B_i^0) with $|A_i| = t$ and $|B_i^0| = s + C$. Then let $\mathcal{Y}_i$ be an inclusion-wise maximal family of pairwise disjoint members of $\mathcal{Y}$ included in B_i^0. Finally, let $B_i = B_i^0 \cap \bigcup \mathcal{Y}_i$ and let H_i be the complete bipartite graph with bipartition (A_i, B_i). Note that the graphs H_i are pairwise edge-disjoint.

Claim 8.4.1. For every $i \in [r]$,

$$\mathbb{P}\left[|B_i| \geq s\right] > 1 - \frac{1}{2n}.$$

Proof of L. et

$$W = B_i^0 \setminus \bigcup \mathcal{Y}_i.$$

We want to show that $\mathbb{P}[|W| > C] < \frac{1}{2n}$.

$$\begin{aligned} &\mathbb{P}[|W| > C] \\ &\quad \leq \mathbb{P}\left[\exists Z \subseteq B_i^0, |Z| = C \text{ and } \forall X \in \binom{Z}{\lceil \frac{p}{2} \rceil}, X \notin \mathcal{Y}\right] \end{aligned}$$

$$\begin{aligned}
&\leq \sum_{Z\in\binom{B_i^0}{C}} \mathbb{P}\left[\forall X \in \binom{Z}{\lceil\frac{p}{2}\rceil}, X \notin \mathcal{Y}\right] \\
&\leq \binom{s+C}{C} \cdot \left(1 - N\cdot\binom{n}{\lceil\frac{p}{2}\rceil}^{-1}\right)^{\binom{C}{\lceil\frac{p}{2}\rceil}} \\
&\leq (s+1)^C \cdot \exp\left(-N\frac{\binom{C}{\lceil\frac{p}{2}\rceil}}{\binom{n}{\lceil\frac{p}{2}\rceil}}\right) \\
&\leq \exp\left(C\ln(s+1) - N\frac{(C-\lceil\frac{p}{2}\rceil)^{\lceil\frac{p}{2}\rceil}}{n^{\lceil\frac{p}{2}\rceil}}\right) \\
&\leq \exp\left(C\ln(s+1) - N2^{-\lceil\frac{p}{2}\rceil}\frac{C^{\lceil\frac{p}{2}\rceil}}{n^{\lceil\frac{p}{2}\rceil}}\right) && \text{since } C \geq 2\left\lceil\frac{p}{2}\right\rceil \\
&\leq \exp\left(C\ln n - N2^{-\lceil\frac{p}{2}\rceil}\frac{C^{\lceil\frac{p}{2}\rceil}}{n^{\lceil\frac{p}{2}\rceil}}\right) && \text{since } s+1 \leq n \\
&\leq \exp\left(n^{1-\frac{1}{2t}}\ln n - N2^{-2\lceil\frac{p}{2}\rceil}\frac{n^{(1-\frac{1}{2t})\lceil\frac{p}{2}\rceil}}{n^{\lceil\frac{p}{2}\rceil}}\right) && \text{since } \frac{1}{2}n^{1-\frac{1}{2t}} \leq C \leq n^{1-\frac{1}{2t}} \\
&\leq \exp\left(n^{1-\frac{1}{2t}}\ln n - 2^{-2\lceil\frac{p}{2}\rceil-1}n^{2-\frac{1}{2t}}\cdot n^{-\frac{1}{2t}\lceil\frac{p}{2}\rceil}\right) && \text{since } \frac{1}{2}n^{2-\frac{1}{2t}} \leq N \\
&\leq \exp\left(n^{1-\frac{1}{2t}}\ln n - 2^{-2\lceil\frac{p}{2}\rceil-1}n^{1-\frac{1}{2t}}\right) && \text{since } \frac{\lceil\frac{p}{2}\rceil}{2t} \leq 1 \\
&= \exp\left(n^{1-\frac{1}{2t}}\left(\ln n - 2^{-2\lceil\frac{p}{2}\rceil-1}n^{\frac{1}{4t}}\right)\right) \\
&< \frac{1}{2n}
\end{aligned}$$

$\diamond$

We deduce from the claim that $\mathbb{P}\left[\forall i \in [r], |B_i| \geq s\right] > 1 - r\frac{1}{2n} \geq \frac{1}{2}$. Observe moreover that $\mathbb{E}[|\mathcal{Y}|] = N$, so by Markov's inequality, $\mathbb{P}[|\mathcal{Y}| > 2N] \leq \frac{1}{2}$. Hence there exists $\mathcal{Y}$ such that $|B_i| \geq s$ for every $i \in [r]$ and $|\mathcal{Y}| \leq 2N$. From now on we fix such a family $\mathcal{Y}$.

For every $i \in [r]$, let

$$\mathcal{X}_i = \{A_i \cup Y \mid Y \in \mathcal{Y}_i\},$$

and let E_i be the set of all the unordered pairs $u, v \in [n]$ such that the number of sets X in $\mathcal{X}_i$ containing both u and v is odd. Observe that for every $i \in [r]$,

$$E(H_i) \subseteq E_i \subseteq E(H_i) \cup \binom{A_i}{2} \cup \bigcup_{Y\in\mathcal{Y}} \binom{Y}{2}.$$

Hence, for every $uv \in E\Delta(\bigcup_{i\in[r]} E_i)$, either $u, v \in Y$ for some $Y \in \mathcal{Y} \cup \{A_i \mid i \in [r]\}$, or $uv \in E \setminus \bigcup_{i\in[r]} E_i$. The point is that this leaves only a few possible pairs of vertices. Let $\mathcal{X}^1$ be the set of all the pairs of the first kind, that it the pairs $uv \in E\Delta(\bigcup_{i\in[r]} E_i)$ with $uv \in E \setminus \bigcup_{i\in[r]} E_i$, and let $\mathcal{X}^2$ be the set of all the $uv \in E\Delta(\bigcup_{i\in[r]} E_i)$ with $u, v \in Y$ for some $Y \in \mathcal{Y} \cup \{A_i \mid i \in [r]\}$.

We take $X_1, \dots, X_\ell$ to be the elements of $\bigcup_{i\in[r]} \mathcal{X}_i$. By construction, we have

$$\ell = \left|\bigcup_{i\in[r]} \mathcal{X}_i\right| \leq \frac{|E|}{\lfloor \frac{p}{2} \rfloor \lceil \frac{p}{2} \rceil}.$$

It remains to bound the size of $\mathcal{X}^1 \cup \mathcal{X}^2$. Since the graphs $H_1, \dots, H_r$ are pairwise edge disjoint, we have $r \leq \frac{|E|}{st}$ and so

$$\begin{aligned} |\mathcal{X}^1| &\leq (t-1)\cdot r \\ &\leq (t-1)\cdot \frac{|E|}{st} \\ &\leq \frac{|E|}{s} \\ &\leq \frac{2n^2}{n^{\frac{1}{2t^2}}} \\ &= 2n^{2-\frac{1}{2t^2}}. \end{aligned}$$

Then, since $G \setminus \bigcup_{i\in[r]} E(H_i)$ does not contain $K_{t,s+C}$ as a subgraph, Theorem 8.4.16 implies

$$\begin{aligned} |\mathcal{X}^1| &\leq 4(s+C)^{\frac{1}{t}} n^{2-\frac{1}{t}} \\ &\leq 4(2n^{1-\frac{1}{2t}})^{\frac{1}{t}} n^{2-\frac{1}{t}} && \text{since } s \leq C \text{ and } C \leq n^{1-\frac{1}{2t}} \\ &\leq 8n^{2-\frac{1}{2t^2}}. \end{aligned}$$

Moreover,

$$\begin{aligned} |\mathcal{X}^2| &\leq \binom{\lceil \frac{p}{2} \rceil}{2} \cdot |\mathcal{Y}| \\ &\leq p^2 \cdot 2N && \text{because } p \geq 2 \text{ and } |\mathcal{Y}| \leq 2N \\ &\leq 2p^2 n^{2-\frac{1}{3t}} && \text{because } N \leq n^{2-\frac{1}{3t}}. \end{aligned}$$

It follows that

$$\begin{aligned} |\mathcal{X}^1 \cup \mathcal{X}^2| &\leq 2n^{2-\frac{1}{2t^2}} + 8n^{2-\frac{1}{2t^2}} + 2p^2 n^{2-\frac{1}{3t}} \\ &\leq \alpha_p n^{2-\epsilon_p} \end{aligned}$$

where $\alpha_p = 2 + 8 + 2p^2$ and $\epsilon_p = \min\{\frac{1}{3t}, \frac{1}{2t^2}\}$. This proves the theorem. □

Corollary 8.4.17. *Let p be an even integer.* $\text{inv}^{=p}(D) \leq \frac{\text{fas}(D)}{\lceil p/2 \rceil \cdot \lfloor p/2 \rfloor} + o(n^2)$ *for every oriented graph D of order n.*

Defining $\text{inv}^{=p}(n)$ as the maximum value of $\text{inv}^{=p}(\overrightarrow{G})$ over all digraphs $\overrightarrow{G}$ of order n, we obtain the following corollary.

Corollary 8.4.18. *If p is an even integer, then* $\text{inv}^{=p}(n) \leq \frac{n}{2\lceil p/2 \rceil \cdot \lfloor p/2 \rfloor} + o(n^2)$.

Once more a natural question is whether this corollary also holds when p is odd.

Problem 8.4.19. *Let $p \geq 3$ be an odd integer. Is is true that* $\text{inv}^{=p}(D) \leq \frac{\text{fas}(D)}{\lceil p/2 \rceil \cdot \lfloor p/2 \rfloor} + o(n^2)$ *for every invertible oriented graph D of order n ?*

8.5 Complexity of computing the ($\leq p$)-inversion number on tournaments

All the restrictions of our problems to instances in which the input oriented graph is a tournament are named with a TOURNAMENT preceding the name of the general problem. For example, this restriction for ($\leq p$)-INVERSION is named TOURNAMENT ($\leq p$)-INVERSION.

8.5.1 NP-completeness of TOURNAMENT ($\leq p$)-INVERSION and TOURNAMENT ($= p$)-INVERSION

Given a tournament T and an integer k, recall that deciding whether T has a feedback arc set of size at most k is NP-complete [Alo06, CTY07]. This means that TOURNAMENT ($\leq p$)-INVERSION and TOURNAMENT ($= p$)-INVERSION are NP-complete when $p = 2$. We show this generalizes to any p, recalling Theorem 8.1.3.

Theorem 8.1.3. TOURNAMENT ($\leq p$)-INVERSION *and* TOURNAMENT ($= p$)-INVERSION *are NP-complete, for any fixed integer p greater than* 2.

Our reductions will be from a hypergraph edge-colouring problem. A *hypergraph* $\mathcal{H}$ consists of a vertex set $V(\mathcal{H})$ and a multiset $\mathcal{E}(\mathcal{H})$ of subsets of $V(\mathcal{H})$, called the *hyperedges* of $\mathcal{H}$. For some nonnegative integer k, we say that $\mathcal{H}$ is *k-uniform* if $|e| = k$ holds for every $e \in \mathcal{E}(\mathcal{H})$ and *k-regular* if every $v \in V(\mathcal{H})$ is contained in exactly k hyperedges. For some positive integer k, a *k-edge-colouring* of a hypergraph $\mathcal{H}$ is a mapping $\phi\colon E(\mathcal{H}) \to [k]$. If $\phi(e) \neq \phi(e')$ holds for each pair of distinct hyperedges e, e' with $e \cap e' \neq \emptyset$, then we say that ϕ is a *proper* k-edge-colouring. We say that a hypergraph is *3-edge-colourable* if it admits a proper 3-edge-colouring.

A hypergraph is *p-special* if it satisfies the following four properties:

(S1) $\mathcal{H}$ is p-uniform;

(S2) $\mathcal{H}$ is 3-regular;

(S3) $|e \cap e'| \leq 1$ for all distinct hyperedges $e, e' \in E(\mathcal{H})$;

(S4) for all triples (v, v', v'') of distinct vertices, if each of $\{v, v'\}, \{v, v''\}$, and $\{v', v''\}$ is contained in a hyperedge of $\mathcal{H}$, then $\{v, v', v''\}$ is contained in a hyperedge of $\mathcal{H}$.

We denote by p-SPECIAL HYPERGRAPH 3-EDGE-COLOURABILITY, in short $(p, 3)$-SHEC, the problem of deciding whether a given p-special hypergraph $\mathcal{H}$ is 3-edge-colourable.

The hardness of $(p, 3)$-SHEC will be crucial for our reduction proving Theorem 8.1.3. Note that the problem $(2, 3)$-SHEC is exactly the problem of deciding whether a cubic, triangle-free graph is 3-edge-colourable. This problem is well known to be NP-hard. (See for example Theorem 4 in [CE91].)

Proposition 8.5.1. $(2, 3)$*-SHEC is NP-hard.*

In the following, we give a slightly technical proof for the fact that this hardness result generalizes for arbitrary $p \geq 3$.

Lemma 8.5.2. $(p, 3)$*-SHEC is NP-hard for every* $p \geq 2$.

Proof. By Proposition 8.5.1, the statement holds for $p = 2$. Now consider an integer $p \geq 3$. Inductively, we may suppose that $(p-1, 3)$-SHEC is NP-hard. We will show the hardness of $(p, 3)$-SHEC by a reduction from $(p-1, 3)$-SHEC.

Let $\mathcal{H}$ be a $(p-1)$-special hypergraph. We now create a hypergraph $\mathcal{H}'$. For every $v \in V(\mathcal{H})$ and every $(i, j) \in [p]^2$, we let $V(\mathcal{H}')$ contain a vertex $v_{i,j}$. Next, for every $e \in E(\mathcal{H})$ and every $(i, j) \in [p]^2$, we let $V(\mathcal{H}')$ contain a vertex $x^e_{i,j}$. For every $e \in E(\mathcal{H})$ and every $(i, j) \in [p]^2$, we let $E(\mathcal{H}')$ contain a hyperedge $e_{i,j} = x^e_{i,j} \cup \{v_{i,j} \mid v \in e\}$. Further, for every $e \in E(\mathcal{H})$ and every $i \in [p]$, we let $E(\mathcal{H}')$ contain a hyperedge $f_{e,i}$ consisting of $\{x^e_{i,j} \mid j \in [p]\}$. Finally, for every $e \in E(\mathcal{H})$ and every $j \in [p]$, we let $E(\mathcal{H}')$ contain a hyperedge $f'_{e,j}$ consisting of $\{x^e_{i,j} \mid i \in [p]\}$. This finishes the description of $\mathcal{H}'$. Observe that $|V(\mathcal{H}')| = p^2(|V(\mathcal{H})| + |E(\mathcal{H})|)$. Then, it is easy to check that $\mathcal{H}'$ is 3-regular because $\mathcal{H}$ is, so we have $|E(\mathcal{H}')| \leq 3|V(\mathcal{H}')|$. Hence, the size of $\mathcal{H}'$ is polynomial in the size of $\mathcal{H}$.

Claim 8.5.1. $\mathcal{H}'$ is p-special.

Proof of W. e need to prove that $\mathcal{H}'$ satisfies (S1), (S2), (S3) and (S4).

It follows immediately from the fact that $\mathcal{H}$ is $(p-1)$-uniform and by construction that $\mathcal{H}'$ satisfies (S1).

We have already argued that $\mathcal{H}'$ is 3-regular, so it satisfies (S2).

We next prove (S3). Let g, g' be distinct hyperedges in $E(\mathcal{H}')$. First suppose that $g = f_{e,i}$ for some $e \in E(\mathcal{H})$ and some $i \in [p]$. If $g' = f_{e',j}$ for some $e' \in E(\mathcal{H})$ and $j \in [p]$, we obtain $(e, i) \neq (e', j)$ by assumption. It follows by construction that $g \cap g' = \emptyset$. Next suppose that $g' = f'_{e',j}$ for some $e' \in E(\mathcal{H})$ and some $j \in [p]$. We have by construction that $g \cap g' = \{x^e_{i,j}\}$ if $e' = e$ and $g \cap g' = \emptyset$, otherwise. Now suppose that $g' = e'_{i',j}$ for some $e' \in E(\mathcal{H})$ and $(i', j) \in [p]^2$. It follows by construction that $g \cap g' = \{x^e_{i,j}\}$ if $(e', i') = (e, i)$, and $g \cap g' = \emptyset$, otherwise. The case of $g = f'_{e,j}$ for some $e \in E(\mathcal{H})$ and some $j \in [p]$ can be handled similarly. We may then suppose that $g = e_{i,j}$ for some $e \in E(\mathcal{H})$ and $(i, j) \in [p]^2$. By symmetry of the roles played by e and e', we may also suppose that $g' = e'_{i',j'}$ for some $e' \in E(\mathcal{H})$ and $(i', j') \in [p]^2$. If $(i, j) \neq (i', j')$, we have $g \cap g' = \emptyset$ by construction. If $(i, j) = (i', j')$, we deduce $g \cap g' = \{v_{i,j} \mid v \in e \cap e'\}$. As $\mathcal{H}$ is $(p-1)$-special, we have $|e \cap e'| \leq 1$ and so $|g \cap g'| \leq 1$. Therefore $\mathcal{H}'$ satisfies (S3).

Let us now prove (S4). Consider three distinct vertices v, v', v'' of $V(\mathcal{H}')$ such that each of $\{v, v'\}$, $\{v, v''\}$, and $\{v', v''\}$ is contained in a hyperedge of $E(\mathcal{H}')$.

First suppose that $v = x^e_{i,j}$ for some $(i, j) \in [p]^2$ and $e \in E(\mathcal{H})$. As $\{v, v'\}$ is contained in a hyperedge of $\mathcal{H}'$, we either have $v' = x^e_{i',j}$ for some $i' \in [p] - i$, $v' = x^e_{i,j'}$ for some $j' \in [p] - j$ or $v' = u'_{i,j}$ for some $u' \in e$. We first deal with the case where $v' = x^e_{i',j}$ for some $i' \in [p] - i$, and discriminate according to the three possibilities for v'' (which are the same as for v').

- If $v'' = x^e_{i'',j}$ for some $i'' \in [p] - \{i, i'\}$, and we obtain $\{v, v', v''\} \subseteq f'_{e,j}$.
- If $v'' = x^e_{i,j''}$ for some $j'' \in [p] - j$, and our construction yields that no hyperedge in $E(\mathcal{H}')$ contains $\{v', v''\}$, a contradiction.
- If $v'' = u''_{i,j}$ for some $u'' \in e$, and again by construction no hyperedge in $E(\mathcal{H}')$ contains $\{v', v''\}$, a contradiction.

Now, up to symmetry between v' and v'', the last remaining case is when $v' = u'_{i,j}$ for some $u' \in e$ and $v'' = u''_{i,j}$ for some $u'' \in e - u'$. Then, we know that $\{v, v', v''\} \subseteq e_{i,j}$ by construction. This concludes the case $v = x^e_{i,j}$, and by symmetry also the cases where $v' = x^{e'}_{i',j'}$ and $v'' = x^{e''}_{i',j'}$.

We may hence suppose that $v = u_{i,j}$ for some $u \in V(\mathcal{H})$ and $(i,j) \in [p]^2$, that $v' = u'_{i',j'}$ for some $u' \in V(\mathcal{H})$ and $(i',j') \in [p]^2$, and that $v'' = u''_{i'',j''}$ for some $u'' \in V(\mathcal{H})$ and $(i'',j'') \in [p]^2$. If $(i,j) \neq (i',j')$, our construction yields that $E(\mathcal{H}')$ does not contain a hyperedge containing $\{v,v'\}$, a contradiction. We therefore obtain $(i',j') = (i,j)$, and symmetrically $(i'',j'') = (i,j)$. By construction, we know each of $\{u,u'\}, \{u,u''\}$, and $\{u',u''\}$ is contained in a hyperedge of $E(\mathcal{H})$. As $\mathcal{H}$ is $(p-1)$-special, we then know there exists some $e \in E(\mathcal{H})$ such that $\{u,u',u''\} \subseteq e$. By construction, it follows that $\{v,v',v''\} \subseteq e_{i,j}$. Therefore, $\mathcal{H}'$ satisfies (S4). ◇

It remains to prove that $\mathcal{H}'$ is 3-edge-colourable if and only if $\mathcal{H}$ is. First suppose that $\mathcal{H}'$ is 3-edge-colourable. Then it admits a proper 3-edge-colouring ϕ'. Let ϕ be the 3-edge-colouring of $\mathcal{H}$ defined by $\phi(e) = \phi'(e_{1,1})$ for every $e \in E(\mathcal{H})$. Let $e, e' \in E(\mathcal{H})$ with $e \cap e' \neq \emptyset$. Then, by construction, we have $e_{1,1} \cap e'_{1,1} \neq \emptyset$. As ϕ' is a proper 3-edge-colouring, $\phi'(e_{1,1}) \neq \phi'(e'_{1,1})$ and hence by construction $\phi(e) \neq \phi(e')$. It follows that ϕ is a proper 3-edge-colouring of $\mathcal{H}$.

Now suppose that $\mathcal{H}$ is 3-edge-colourable. Then it admits a proper 3-edge-colouring ϕ. We now define a 3-edge-colouring ϕ' of $\mathcal{H}'$ as follows. First, for every $e \in E(\mathcal{H})$ and $(i,j) \in [p]^2$, we set $\phi'(e_{i,j}) = \phi(e)$. Now consider some $e \in E(\mathcal{H})$ and let α, β be the unique integers with $\alpha < \beta$ and $\{\alpha, \beta\} = [3] \setminus \{\phi(e)\}$. We then set $\phi'(f_{e,i}) = \alpha$ for all $i \in [p]$ and $\phi'(f'_{e,j}) = \beta$ for all $j \in [p]$. This finishes the description of ϕ'. Let us prove that ϕ' is proper. Consider some $g, g' \in E(\mathcal{H}')$ with $g \cap g' \neq \emptyset$. If $g = f_{e,i}$ for some $e \in E(\mathcal{H})$ and some $i \in [p]$ or $g = f'_{e,j}$ for some $e \in E(\mathcal{H})$ and some $j \in [p]$, we obtain $\phi'(g) \neq \phi'(g')$ by construction of ϕ'. We may hence suppose that $g = e_{i,j}$ for some $e \in E(\mathcal{H})$ and some $(i,j) \in [p]^2$. Symmetrically, we may suppose that $g' = e'_{i',j'}$ for some $e' \in E(\mathcal{H})$ and some $(i',j') \in [p]^2$. If $(i,j) \neq (i',j')$, we have $g \cap g' = \emptyset$ by construction, a contradiction. We then have $(i',j') = (i,j)$ and $e \neq e'$. By construction and as $g \cap g' \neq \emptyset$, we then have $e \cap e' \neq \emptyset$. As ϕ is a proper 3-edge-colouring of $\mathcal{H}$, we have $\phi(e) \neq \phi(e')$ and hence by construction that $\phi'(g) \neq \phi'(g')$. Therefore ϕ' is a proper 3-edge-colouring of $\mathcal{H}'$. This finishes the proof. □

For the main part of the proof of Theorem 8.1.3, we further need the following simple observation.

Proposition 8.5.3. *Let D be a digraph and $\mathcal{X}$ be a decycling $(\leq p)$-family of size k for some positive integers k and p. If there exists an arc $uv \in A(D)$ and pk directed cycles $C_1, \dots, C_{pk}$ in D such that $V(C_i) \cap V(C_j) = \{u,v\}$ for all distinct $i,j \in [pk]$ and $uv \in A(C_i)$ for all $i \in [pk]$, then there exists $X \in \mathcal{X}$ such that $\{u,v\} \subseteq X$.*

Proof. Set $Y = \bigcup_{X \in \mathcal{X}} X$, and observe that $|Y| \leq kp$. For every $i \in [pk]$, at least one arc of C_i is spanned by Y, so there exists $X_i \in \mathcal{X}$ containing at least two vertices of C_i. If none of the X_i contains $\{u,v\}$, then each X_i contains at least one vertex in $V(C_i) \setminus \{u,v\}$. Now, either some X_i contains two elements of $V(C_i) \setminus \{u,v\}$, or it contains u or v, and in any case we have $|Y| \geq kp+1$, a contradiction. Thus one of the X_i contains $\{u,v\}$. □

We shall also use the following notation : given two disjoint sets of vertices A and B in a digraph D, we write $A \Rightarrow B$ if all possibles arcs with tail in A and head in B. We are now ready to proceed to the main proof of Theorem 8.1.3.

Proof of Theorem 8.1.3. Let us prove the NP-hardness of TOURNAMENT $(\leq p)$-INVERSION.

The proof is a reduction from $(p-1,3)$-SHEC to TOURNAMENT $(\leq p)$-INVERSION. Let $\mathcal{H}$ be a $(p-1)$-special hypergraph and let $(v_1, \dots, v_n)$ be an arbitrary ordering of $V(\mathcal{H})$. We further set $k = |E(\mathcal{H})|$. We now create a tournament T. We first let $V(T)$ contain $V(\mathcal{H})$. Further, for all distinct $i, j \in [n]$ with $i < j$, we let $A(T)$ contain (v_j, v_i) if $\{v_i, v_j\} \subseteq e$ for some $e \in E(\mathcal{H})$, and (v_i, v_j), otherwise. Next, for $i \in [n]$, we let $V(T)$ contain a set W_i of pk vertices, and we let $A(T)$ contain arcs such that $T[W_i]$ is a transitive tournament. We set $W = \bigcup_{i\in[n]} W_i$. Next, for $i \in [n]$, we let $\{v_1, \ldots, v_{i-1}\} \Rightarrow W_i$ and $W_i \Rightarrow \{v_i, \ldots, v_n\}$ in T. Further, for all distinct $i, j \in [n]$, with $i < j$ we let $W_i \Rightarrow W_j$ in T. Finally, we let $V(T)$ contain a set $Z = \{z_1, z_2, z_3\}$. We let $Z \Rightarrow W$ and $V(\mathcal{H}) \Rightarrow Z$ in T. Finally, we add some arcs such that $T[Z]$ is a transitive tournament. This finishes the description of T. Observe that the size of T is polynomial in the size of $\mathcal{H}$.

Let us prove that $\text{inv}^{\leq p}(T) \leq k$ if and only if $\mathcal{H}$ is 3-edge-colourable.

First suppose that $\mathcal{H}$ admits a proper 3-edge-colouring ϕ. For every $e \in E(H)$, we set $X_e = e \cup \{z_{\phi(e)}\}$. Observe that $|X_e| = p$, as $\mathcal{H}$ is $(p-1)$-uniform. Set $\mathcal{X} = (X_e \mid e \in E(\mathcal{H}))$ and $X = \bigcup_{e\in E(\mathcal{H})} X_e$. Observe that $\mathcal{X}$ contains exactly k sets. Now let $T' = \text{Inv}(T; \mathcal{X})$. We shall now prove that T' is a transitive tournament.

First consider some $i \in [n]$ and some $j \in [3]$. Then, by construction, we know $A(T)$ contains the arc $v_i z_j$. As $\mathcal{H}$ is 3-regular and ϕ is a proper 3-edge-colouring of $\mathcal{H}$, there exists exactly one $e \in E(\mathcal{H})$ with $v_i \in e$ and $\phi(e) = j$. It follows that $\{v_i, z_j\} \subseteq X_e$ and $\{v_i, z_j\} \nsubseteq X_{e'}$ for all $e' \in E(\mathcal{H}) \setminus \{e\}$. It follows that $A(T')$ contains the arc (z_j, v_i). Hence $Z \Rightarrow V(\mathcal{H})$ in T'. Furthermore, as $X \cap W = \emptyset$, we have $Z \Rightarrow W$ in T'. Next, as $|X_e \cap Z| \leq 1$ for all $e \in E(\mathcal{H})$, we have $T'[Z] = T[Z]$, and so $T'[Z]$ is a transitive tournament. It hence suffices to prove that $T' - Z$ is a transitive tournament.

To this end, consider some distinct $i, j \in [n]$ with $i < j$. If there exists no hyperedge in $E(\mathcal{H})$ containing $\{v_i, v_j\}$, then, by construction, we know $A(T)$ contains (v_i, v_j) and $\{v_i, v_j\} \setminus X_e \neq \emptyset$ for all $e \in E(\mathcal{H})$. It follows that $A(T')$ contains the arc (v_i, v_j). If there exists a hyperedge $e \in E(\mathcal{H})$ containing $\{v_i, v_j\}$, then, by the property (S3) of $(p-1)$-special hypergraph, e is the only hyperedge containing $\{v_i, v_j\}$. It follows by construction that $A(T)$ contains (v_j, v_i), and as $\{v_i, v_j\} \subseteq X_e$, $\{v_i, v_j\} \nsubseteq X_{e'}$ for all $e' \in E(\mathcal{H}) \setminus \{e\}$, we deduce that $A(T')$ contains (v_i, v_j). Therefore $T'[V(H)]$ is transitive with acyclic ordering $(v_1, \ldots, v_n)$. Now since we have $X \cap W = \emptyset$, all the arcs incident to a vertex of W are unchanged we inverting the sets of $\mathcal{X}$. Hence, in T', $\{v_1, \ldots, v_{i-1}\} \Rightarrow W_i$ and $W_i \Rightarrow \{v_i, \ldots, v_n\}$ for all $i \in [n]$, $W_i \Rightarrow W_j$ for all distinct $i, j \in [n]$. Therefore, $T' - Z$ is a transitive tournament, which implies that T' is a transitive tournament.

Now suppose that T has a decycling $(\leq p)$-family $\mathcal{X}$ of size at most k. For $i \in [k]$, let F_i be the set of arcs of $A(T)$ for which there exist exactly i sets in $\mathcal{X}$ each of which contain both endvertices of the arc and let $F = \bigcup_{i\in[k]} F_i$. Further, let A_1 be the set of all arcs oriented from $V(\mathcal{H})$ to Z in $A(T)$, and A_2 those arcs in $A(T)$ of the form (v_j, v_i) for some $i, j \in [n]$ with $i < j$.

Claim 8.5.2. $F = F_1 = A_1 \cup A_2$ and $|X| = p$ for all $X \in \mathcal{X}$.

Proof of F. irst consider an arc $v_i z_j \in A_1$ for some $i \in [n]$ and $j \in [3]$. Observe that for every $w \in W_1$, (z_j, w, v_i) is a triangle in T, then Proposition 8.5.3 and the fact that $|W_1| = pk$ yields $v_i z_j \in F$, therefore $A_1 \subseteq F$.

Next consider some distinct $i, j \in [n]$ with $i < j$ such that $(v_j, v_i) \in A_2$. Observe that for every $w \in W_j$, (v_i, w, v_j, v_i) is a directed 3-cycle in T. Hence, $(v_j, v_i) \in F$ by Proposition 8.5.3, since $|W_1| = pk$, therefore $A_2 \subseteq F$.

As $A_1 \cap A_2 = \emptyset$, we get $|F| \geq |A_1| + |A_2|$. Moreover as $\mathcal{H}$ is $(p-1)$-special, a double counting of the vertex-hyperedge incidences yields $3n = k(p-1)$, therefore:

$$\begin{aligned}
k\binom{p}{2} &\geq \sum_{X\in\mathcal{X}} \binom{|X|}{2} \\
&= \sum_{i=1}^{k} i|F_i| \\
&\geq |F| \\
&\geq |A_1| + |A_2| \\
&= 3n + k\binom{p-1}{2} \\
&= k(p-1) + k\binom{p-1}{2} \\
&= k\binom{p}{2}.
\end{aligned}$$

Then, equality holds throughout and the statement follows. ◇

Now consider some $X \in \mathcal{X}$. Then, Claim 8.5.2, yields that $|X| = p$, $X \subseteq Z \cup V(\mathcal{H})$ and $F \cap A(T[Z]) = \emptyset$, hence $|X \cap Z| \leq 1$. Now consider some distinct $i, j \in [n]$ such that $\{v_i, v_j\} \subseteq X$. By Claim 8.5.2, one of (v_i, v_j) and (v_j, v_i) is contained in A_2, and, by construction and (S3), there exists a unique hyperedge $e \in E(\mathcal{H})$ such that $\{v_i, v_j\} \subseteq e$. As v_i and v_j were chosen to be arbitrary elements of $X \cap V(\mathcal{H})$ and by the property (S4), we have $X \cap V(\mathcal{H}) \subseteq e$. We now obtain $p = |X| = |X \cap Z| + |X \cap V(\mathcal{H})| \leq 1 + |e| = 1 + (p-1) = p$, hence equality holds throughout which yields $X = e \cup z_j$ for some $j \in [3]$. As $F = F_1$ by Claim 8.5.2, for every $e \in E(\mathcal{H})$, there exists at most one $X \in \mathcal{X}$ with $e \subseteq X$. Then, since $|\mathcal{X}| = k = |E(\mathcal{H})|$, for every $e \in E(\mathcal{H})$, there exists one (and exactly one) $j_e \in [3]$ such that $e \cup z_{j_e} \in \mathcal{X}$.

We now define a 3-edge-colouring ϕ of $\mathcal{H}$ by setting $\phi(e) = j_e$, and let us prove that it is proper. Consider some $v \in V(\mathcal{H})$ and suppose for the sake of a contradiction that there exist two distinct hyperedges $e, e' \in E(\mathcal{H})$ containing v with $\phi(e) = \phi(e')$. It follows by construction that $\mathcal{X}$ contains both the sets $e \cup z_{\phi(e)}$ and $e' \cup z_{\phi(e)}$. We then have $vz_{\phi(e)} \in F \setminus F_1$, a contradiction to Claim 8.5.2.

The NP-hardness proof for Tournament $(= p)$-Inversion is very similar to the one for Tournament $(\leq p)$-Inversion. Indeed, instances created in the reduction are positive for Tournament $(\leq p)$-Inversion if and only if they are positive for $(p-1, 3)$-SHEC if and only if they are positive for Tournament $(= p)$-Inversion. That is because for any produced instance, both directions hold: firstly, the decycling family provided by the solution to $(p-1, 3)$-SHEC is clearly a $(= p)$-family by definition; secondly, if there is a decycling $(= p)$-family of size k, it is also a $(\leq p)$-family, and we have shown that it corresponds to a positive instance of $(p-1, 3)$-SHEC. Actually the proof for $(= p)$-inversion is a subproof of the current one in the sense that we do not need Claim 8.5.2 to justify that each inversion has size p. □

8.5.2 A $O(k^2p^3)$ kernel for Tournament Bounded Size Inversion

In this section, we study the complexity of the two following versions of $(\leq p)$-Inversion and $(= p)$-Inversion, which differ only by including p in the input. This allows us to parameterize by both the solution size k, and p, which enables us to derive a polynomial kernel (in p and k) for tournaments.

BOUNDED SIZE INVERSION
Input: An oriented graph D and two positive integers p and k
Question: Does D admit a decycling $(\leq p)$-family of size at most k?

PRESCRIBED SIZE INVERSION
Input: An oriented graph D and two positive integers p and k
Question: Does D admit a decycling $(= p)$-family of size at most k?

Recall that while FEEDBACK ARC SET IN TOURNAMENTS is NP-complete, it admits a $(2 + \epsilon)k$ kernel for every $\epsilon > 0$ thanks to Bessy et al. [Bes+11], using the PTAS of Mathieu and Schudy [KS07]. In our language, this means it is also a kernel for 2-inversion. Here, we consider instances (T, p, k) for $(\leq p)$-inversion, and show a polynomial kernel in p and k. Our algorithm also leverages the PTAS of [KS07] to find a feedback arc set F of size bounded by a function of p and k. Then, $T \setminus F$ is acyclic, and we consider the positions of the arcs of F in T to prune vertices "in between" while preserving an equivalent instance, as long as the "gaps" are large enough.

Theorem 8.5.4. *For any $\epsilon > 0$,* TOURNAMENT BOUNDED SIZE INVERSION *admits a kernel of size* $(1 + \epsilon)k^2p^3 + o(k^2p^3)$.

Proof. Consider an instance (T, p, k) for TOURNAMENT BOUNDED SIZE INVERSION, with tournament T and integers k, p, and fix some $\epsilon > 0$. We describe a polynomial-time algorithm producing an instance (T', p, k) equivalent to (T, p, k), such that $|T'| = O(k^2p^3)$.

Note that any $(\leq p)$-inversion reverses the direction of at most $\binom{p}{2}$ arcs. Thus, if (T, p, k) is a positive instance for TOURNAMENT BOUNDED SIZE INVERSION, then T admits a feedback arc set of size at most $M \leq k\binom{p}{2}$. We first run the PTAS of [KS07] to yield a feedback arc set F such that $|F| \leq (1+\epsilon)M$. Now, if $|F| > (1+\epsilon)k\binom{p}{2}$, we may output a trivial NO instance (such as a directed 3-cycle with $k = 0$). We may hence suppose that $|F| \leq (1 + \epsilon)k\binom{p}{2}$ in the following. Then, we can assume F is arc-minimal, because an arc of F the removal of which still yields a feedback arc set can be found and removed in quadratic time.

We then order $V(T)$ according to an acyclic ordering σ of $T \setminus F$, as a mapping $\sigma : V(T) \to |V(T)|$, which can be obtained in polynomial time. Since F is arc-minimal, note that any $(a, b) \in F$ is such that $\sigma(b) < \sigma(a)$. Let $S \subseteq V(T)$ be the set of endpoints of arcs in F, then $|S| \leq 2(1+\epsilon)k\binom{p}{2}$. A set $I \subseteq V(T)$ is called an *interval* of $(V(T), \sigma)$ if $I = \sigma^{-1}([i, j])$ for some $i < j \leq |V(T)|$. We employ the following reduction rule, running in polynomial time, as long as it applies:

♣ Take an arbitrary interval I of $(V(T), \sigma)$, disjoint from S, and containing more than $kp + 1$ vertices, then delete an arbitrary vertex $z \in I$.

Claim 8.5.3. The instance (T', p, k), output after one application of ♣ is a YES instance if and only if (T, p, k) is.

Proof of the claim. First, if (T, p, k) admits a solution of size (at most) k, say $X_1, ..., X_k$, then letting $X'_i = X_i - z$ for each $i \in [k]$ yields that $X'_1, ..., X'_k$ is a solution to (T', p, k).

Now, assume (T', p, k) admits a solution of size (at most) k, say $(X'_1, \ldots, X'_k)$, resulting in the transitive tournament R'. Consider inversions $X'_1 \setminus I, \ldots, X'_k \setminus I$ on T, resulting in tournament R, and let us show that they are a solution of size k. Assume for the sake of contradiction that R is not acyclic and hence, as it is a tournament, contains a triangle $C = (a, b, c)$. This triangle necessarily involves a vertex in I, say it is b (possibly equal to z). Indeed, $R[V(T) \setminus I]$ is equal to the result of $(X'_1 \setminus I, \dots, X'_k \setminus I)$ applied to $T[V(T) \setminus I]$, which is also the transitive tournament $R'[V(T) \setminus I]$. Now, $b \in I$, and b is not part of any inversions $X'_1 \setminus I, \dots, X'_k \setminus I$, thus arcs (a, b) and (b, c) existed in $T \setminus F$, which is acyclically ordered by σ, yielding $\sigma(a) < \sigma(b) < \sigma(c)$. We argue that none of a or c belongs to I. Indeed, otherwise arc (c, a) would not have been inverted by any $X'_i \setminus I$. In that case, (c, a) was also an arc of T, but cannot be an arc of $T \setminus F$ since $\sigma(a) < \sigma(c)$, so $(c, a) \in F \subseteq S^2$, which is absurd. Therefore, (c, a) is also an arc in R', because $R'[V(T) \setminus I] = R[V(T) \setminus I]$. Consider T', and take any $z' \in I - z$ that does not belong to any $(X'_1, \dots, X'_k)$, which is possible since $|I - z| \geq kp + 1$. Observe that as $T \setminus F$ is acyclic and acyclically ordered by σ, the fact that $a, c \notin I$, $b \in I$, and $\sigma(a) < \sigma(b) < \sigma(c)$ yields that $\sigma(a) < \sigma(z') < \sigma(c)$ as well. Arcs (a, z') and (z', b) exist in $T \setminus F$, as $z' \notin S$, and since z' does not belong to any X'_i, these arcs must also belong to R', yielding cycle (a, z, c), which is absurd and concludes the equivalence. ◇

We apply rule ♣ repeatedly, while the condition holds, ending in the equivalent instance (T', k, p). At this point, T' still contains the vertices of S, with $|S| \leq 2(1 + \epsilon)k\binom{p}{2}$. Furthermore, $V(T) \setminus S$ can be partitioned into a collection of $2(1 + \epsilon)k\binom{p}{2} + 1$ intervals with respect to σ, each of which contains at most $kp + 1$ vertices. This yields that $|V(T')| \leq 2(1 + \epsilon)k\binom{p}{2} + (kp + 1)\left(2(1 + \epsilon)k\binom{p}{2} + 1\right)$, and in turn $|V(T')| \leq (1 + \epsilon)k^2p^3 + o(k^2p^3)$. □

8.5.3 W[1]-hardness of Bounded Size Inversion parameterized by p.

We recall the Multicoloured Clique problem, a common restriction of Clique, which is still $W[1]$-hard, but allows for easier reductions in many hardness proofs.

Multicoloured Clique (MCC)
Parameter: k
Input: A graph G, along with a partition $(V_1, \dots, V_k)$ of $V(G)$
Question: Does G contain a clique with exactly one vertex from each V_i?

The hardness of MCC derives by a straightforward parameterized reduction from k-Clique, shown by Fellows et al. [Fel+09].

Proposition 8.5.5 (Fellows et al. '09)**.** MultiColoured Clique *is W[1]-hard.*

We are now ready to show the hardness of Bounded Size Inversion parameterized by p, reducing from MCC.

Theorem 8.1.5. *Given a digraph D, both deciding whether D can be made acyclic by one $(\leq p)$-inversion, and deciding whether D can be made acyclic by one $(= p)$-inversion, are $W[1]$-hard parameterized by p.*

Proof. We prove this statement for one $(\leq p)$-inversion by a reduction from MCC, then argue the reduction also holds for one $(= p)$-inversion.

Consider an instance $(G, (V_1, \dots, V_k))$ of Multicoloured Clique, and let us construct an instance (D, p) to the $(\leq p)$-inversion problem. For $i \in [k]$, let $q_i = |V_i|$ and let $v_i^1, \dots, v_i^{q_i}$ be an arbitrary enumeration of V_i. For every $i \in [k]$ and $j \in [q_i]$, we let $V(D)$ contain two vertices w_i^j and x_i^j and for every $i \in [k]$, we let $A(D)$ contain arcs such that $w_i^1 x_i^1 \dots w_i^{q_i} x_i^{q_i} w_i^1$ is a cycle which we denote by C_i. Further, we denote by F the set of all ordered pairs $(v_{i_1}^{j_1}, v_{i_2}^{j_2})$ with $1 \leq i_1 < i_2 \leq k, j_1 \in [q_{i_1}], j_2 \in [q_{i_2}]$ and such that $E(G)$ does *not* contain an edge linking $v_{i_1}^{j_1}$ and $v_{i_2}^{j_2}$. We let $V(D)$ contain a set Z that contains a vertex z_f for every $f \in F$. Further, for every $f = (v_{i_1}^{j_1}, v_{i_2}^{j_2}) \in F$, we let $A(D)$ contain the arcs $w_{i_1}^{j_1} z_f, z_f w_{i_2}^{j_2}$, and $w_{i_1}^{j_1} w_{i_2}^{j_2}$. Finally, we set $p = 2k$. This finishes the description of (D, p). For an illustration, see Figure 8.1.

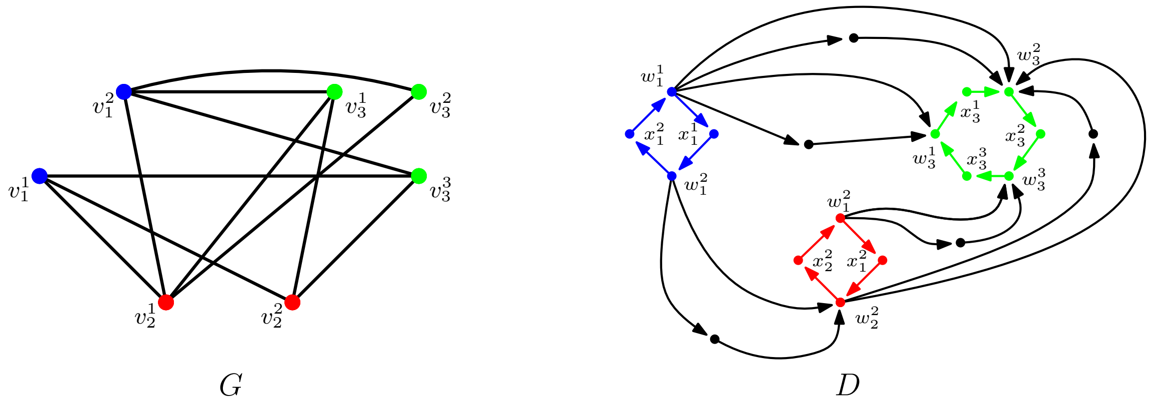


Figure 8.1: An illustration of an instance $(G, (V_1, V_2, V_3))$ of MCC and the corresponding digraph D. The vertices of V_1, V_2, and V_3 are marked in blue, red, and green in G, respectively. Here, non-edges of G yield $F = (v_1^1, v_3^1), (v_1^1, v_3^2), (v_1^2, v_2^2), (v_2^1, v_3^3), (v_2^2, v_3^2)$. For $i \in [3]$, the directed cycle C_i in D is marked in the same colour as V_i in G. All vertices of Z and arcs of D incident to Z are marked in black.

It is easy to see that (D, p) can be constructed in polynomial time from $(G, (V_1, \dots, V_k))$ and clearly, p is bounded by a computational function of k. It hence suffices to prove that $\text{inv}^{\leq p}(D) \leq 1$ if and only if $(G, (V_1, \dots, V_k))$ is a Yes instance of MCC.

First suppose that $(G, (V_1, \dots, V_k))$ is a Yes instance of MCC, so for $i \in [k]$, there exists some $r_i \in [q_i]$ such that $G[\{v_1^{r_1}, \dots, v_k^{r_k}\}]$ is a clique. Set $X = \{w_1^{r_1}, x_1^{r_1}, \dots, w_k^{r_k}, x_k^{r_k}\}$ and $D' = \text{Inv}(D, X)$. Clearly, we have $|X| = 2k = p$. It now suffices to prove that D' is acyclic. In D, we have $V(D) = Z \cup \bigcup_{i \in [k]} C_i$, and its non-trivial strongly connected components are exactly the directed cycles $(C_i)_i$. Now, as $\{v_1^{r_1}, \dots, v_k^{r_k}\}$ induces a clique in G, the set X must induce a matching $(w_i^{r_i}, x_i^{r_i})_{i \in [k]}$ in D. For any i, the inversion of $(w_i^{r_i}, x_i^{r_i})$ yields that $D'[C_i]$ is acyclic. Moreover, note that any vertices joined by a path in D' must have also been joined in D. Indeed, any path of D' using the arc $(w_i^{r_i}, x_i^{r_i})$ may be replaced by the directed path from $w_i^{r_i}$ to $x_i^{r_i}$ along the other side of C_i. In particular, any directed cycle in D' involves only vertices contained in a strongly connected component of D, so it must be contained in some C_i, which is absurd. Therefore, $\text{inv}^{\leq p}(D) \leq 1$.

Now suppose that $\text{inv}^{\leq p}(D) \leq 1$, so there exists a $(\leq p)$-set $X \subseteq V(D)$ such that D' is acyclic where $D' = \text{Inv}(D, X)$. As C_i is a directed cycle for $i \in [k]$ and $V(C_{i_1}) \cap V(C_{i_2}) = \emptyset$ for all distinct $i_1, i_2 \in [k]$, we have $2k = p \geq |X| = \sum_{i=1}^{k} |X \cap V(C_i)| + |X \cap Z| \geq \sum_{i=1}^{k} |X \cap V(C_i)| \geq \sum_{i=1}^{k} 2 = 2k$. Then, the equality holds throughout and we have $X \cap Z = \emptyset$ and $|X \cap V(C_i)| = 2$ for $i \in [k]$. Now consider some $i \in [k]$. As C_i is a directed cycle and D' is acyclic, X contains two consecutive vertices of C_i. As $|V(C_i) \cap X| = 2$, this yields that there exists one unique $r_i \in [q_i]$ such that

$w_i^{r_i} \in X$.

We claim that $G[\{v_1^{r_1}, \dots, v_k^{r_k}\}]$ is a clique. Suppose otherwise, so there exists some distinct $i_1, i_2 \in [k]$ with $i_1 < i_2$ such that $E(G)$ does not contain an edge linking $v_{i_1}^{r_{i_1}}$ and $v_{i_2}^{r_{i_2}}$. It follows that $f = (v_{i_1}^{r_{i_1}}, v_{i_2}^{r_{i_2}}) \in F$. As $\{w_{i_1}^{r_{i_1}}, w_{i_2}^{r_{i_2}}\} \subseteq X$ and $X \cap Z = \emptyset$, we now know that $w_{i_1}^{r_{i_1}} z_p w_{i_2}^{r_{i_2}} w_{i_1}^{r_{i_1}}$ is directed cycle in D', a contradiction. Therefore, $G[\{v_1^{r_1}, \dots, v_k^{r_k}\}]$ is a clique and so $(G, (V_1, \dots, V_k))$ is a YES instance of MCC.

The same reduction applies when considering one $(= p)$-inversion, again taking $p = 2k$. Indeed, for a solution X to MCC, the inversion set we produce has size exactly $p = 2k$. Conversely, recall that in a reduced instance, the k directed cycles $(C_i)_i$ share no vertices, so in a positive instance, the solution contains at least two vertices in each, so it has size exactly $2k$. □

8.6 Further directions

While we have shown BOUNDED SIZE INVERSION is not FPT parameterized by the size of the inversions, the question of obtaining an FPT algorithm parameterized by k for every fixed p remains open.

Problem 8.6.1. *Does BOUNDED SIZE INVERSION admit an algorithm running in $f(k)n^{O(f(p))}$ time?*

Going back to the problem of inversions without size constraints, where TOURNAMENT k-INVERSION is FPT, it would be interesting to obtain a generalization of Theorem 8.5.4.

Problem 8.6.2. *Does TOURNAMENT k-INVERSION admit a polynomial kernel?*

Bibliography

[Abo25] Pierre Aboulker. *Personal Communication.* 2025.

[AA23] Pierre Aboulker and Guillaume Aubian. "Four Proofs of the Directed Brooks' Theorem". In: *Discrete Mathematics.* Special Issue in Honour of Landon Rabern 346.11 (Nov. 2023), p. 113193. DOI: 10.1016/j.disc.2022.113193.

[AAC23] Pierre Aboulker, Guillaume Aubian, and Pierre Charbit. *Digraph Colouring and Arc-Connectivity.* Sept. 2023. DOI: 10.48550/arXiv.2304.04690. arXiv: 2304.04690 [math].

[AAC24] Pierre Aboulker, Guillaume Aubian, and Pierre Charbit. "Heroes in Oriented Complete Multipartite Graphs". In: *Journal of Graph Theory* 105.4 (2024), pp. 652–669. DOI: 10.1002/jgt.23061.

[Abo+23] Pierre Aboulker, Guillaume Aubian, Pierre Charbit, and Raul Lopes. *Clique Number of Tournaments.* Oct. 2023. DOI: 10.48550/arXiv.2310.04265. arXiv: 2310.04265 [cs, math].

[Abo+24] Pierre Aboulker, Guillaume Aubian, Pierre Charbit, and Stéphan Thomassé. "(P6, Triangle)-Free Digraphs Have Bounded Dichromatic Number". In: *The Electronic Journal of Combinatorics* (Dec. 2024), P4.60–P4.60. DOI: 10.37236/11838.

[AAS22] Pierre Aboulker, Guillaume Aubian, and Raphael Steiner. "Heroes in Orientations of Chordal Graphs". In: *SIAM Journal on Discrete Mathematics* 36.4 (Dec. 2022), pp. 2497–2505. DOI: 10.1137/22M1481427.

[ABV22] Pierre Aboulker, Nicolas Bousquet, and Rémi de Verclos. "Chordal Directed Graphs Are Not χ-Bounded". In: *The Electronic Journal of Combinatorics* (May 2022), P2.17–P2.17. DOI: 10.37236/11050.

[Abo+17] Pierre Aboulker, Nick Brettell, Frédéric Havet, Dániel Marx, and Nicolas Trotignon. "Coloring Graphs with Constraints on Connectivity". In: *Journal of Graph Theory* 85.4 (2017), pp. 814–838. DOI: 10.1002/jgt.22109.

[ACN21] Pierre Aboulker, Pierre Charbit, and Reza Naserasr. "Extension of Gyárfás-Sumner Conjecture to Digraphs". In: *The Electronic Journal of Combinatorics* (May 2021), P2.27–P2.27. DOI: 10.37236/9906.

[Abo+19] Pierre Aboulker, Nathann Cohen, Frédéric Havet, William Lochet, Phablo F. S. Moura, and Stéphan Thomassé. "Subdivisions in Digraphs of Large Out-Degree or Large Dichromatic Number". In: *The Electronic Journal of Combinatorics* (July 2019), P3.19–P3.19. DOI: 10.37236/6521.

[AHT07] L. Addario-Berry, F. Havet, and S. Thomassé. “Paths with Two Blocks in N-Chromatic Digraphs”. In: *Journal of Combinatorial Theory, Series B* 97.4 (July 2007), pp. 620–626. DOI: 10.1016/j.jctb.2006.10.001.

[Add+13] Louigi Addario-Berry, Frédéric Havet, Cláudia Linhares Sales, Bruce Reed, and Stéphan Thomassé. “Oriented Trees in Digraphs”. In: *Discrete Mathematics* 313.8 (Apr. 2013), pp. 967–974. DOI: 10.1016/j.disc.2013.01.011.

[Aki+19] Hugo A. Akitaya, Rajasekhar Inkulu, Torrie L. Nichols, Diane L. Souvaine, Csaba D. Tóth, and Charles R. Winston. “Minimum Weight Connectivity Augmentation for Planar Straight-Line Graphs”. In: *Theoretical Computer Science*. Selected Papers from the 11th International Conference and Workshops on Algorithms and Computation 789 (Oct. 2019), pp. 50–63. DOI: 10.1016/j.tcs.2018.05.031.

[AK00] Paola Alimonti and Viggo Kann. “Some APX-completeness Results for Cubic Graphs”. In: *Theoretical Computer Science* 237.1-2 (Apr. 2000), pp. 123–134. DOI: 10.1016/S0304-3975(98)00158-3.

[Alo96] Noga Alon. “Disjoint Directed Cycles”. In: *Journal of Combinatorial Theory, Series B* 68.2 (Nov. 1996), pp. 167–178. DOI: 10.1006/jctb.1996.0062.

[Alo06] Noga Alon. “Ranking Tournaments”. In: *SIAM Journal on Discrete Mathematics* 20.1 (Jan. 2006), pp. 137–142. DOI: 10.1137/050623905.

[APS01] Noga Alon, János Pach, and József Solymosi. “Ramsey-Type Theorems with Forbidden Subgraphs”. In: *Combinatorica* 21.2 (Apr. 2001), pp. 155–170. DOI: 10.1007/s004930100016.

[Alo+24] Noga Alon, Emil Powierski, Michael Savery, Alex Scott, and Elizabeth Wilmer. “Invertibility of Digraphs and Tournaments”. In: *SIAM Journal on Discrete Mathematics* 38.1 (Mar. 2024), pp. 327–347. DOI: 10.1137/23M1547135.

[AH76] K. Appel and W. Haken. “Every Planar Map Is Four Colorable”. In: *Bull. Amer. Math. Soc.* 82.3 (1976), pp. 711–712. DOI: 10.1090/S0002-9904-1976-14122-5.

[AG60] E. Asplund and B. Grünbaum. “On a Coloring Problem.” In: *MATHEMATICA SCANDINAVICA* 8 (Dec. 1960), pp. 181–188. DOI: 10.7146/math.scand.a-10607.

[ACK19] Sepehr Assadi, Yu Chen, and Sanjeev Khanna. “Sublinear Algorithms for (? + 1) Vertex Coloring”. In: *Proceedings of the 2019 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. Proceedings. Society for Industrial and Applied Mathematics, Jan. 2019, pp. 767–786. DOI: 10.1137/1.9781611975482.48.

[Aub+25] Guillaume Aubian, Frédéric Havet, Florian Hörsch, Felix Kingelhoefer, Nicolas Nisse, Clément Rambaud, and Quentin Vermande. “Problems, Proofs, and Disproofs on the Inversion Number”. In: *The Electronic Journal of Combinatorics* (Mar. 2025), P1.42–P1.42. DOI: 10.37236/12983.

[BW14] Bradley Baetz and David R. Wood. *Brooks' Vertex-Colouring Theorem in Linear Time*. Jan. 2014. DOI: 10.48550/arXiv.1401.8023. arXiv: 1401.8023 [cs].

[BBT14] Jørgen Bang-Jensen, Stéphane Bessy, and Stéphan Thomassé. “Disjoint 3-Cycles in Tournaments: A Proof of The Bermond–Thomassen Conjecture for Tournaments”. In: *Journal of Graph Theory* 75.3 (2014), pp. 284–302. DOI: 10.1002/jgt.21740.

[BG02] Jørgen Bang-Jensen and Gregory Gutin. *Digraphs*. London: Springer, 2002. DOI: 10.1007/978-1-4471-3886-0.

[BG09] Jørgen Bang-Jensen and Gregory Z. Gutin. *Digraphs*. Springer Monographs in Mathematics. London: Springer London, 2009. DOI: 10.1007/978-1-84800-998-1.

[Ban+25] Jørgen Bang-Jensen, Frédéric Havet, Florian Hörsch, Clément Rambaud, Amadeus Reinald, and Caroline Silva. *Making an Oriented Graph Acyclic Using Inversions of Bounded or Prescribed Size*. Nov. 2025. DOI: 10.48550/arXiv.2511.22562. arXiv: 2511.22562 [math].

[BJ98] Jørgen Bang-Jensen and Tibor Jordán. "Edge-Connectivity Augmentation Preserving Simplicity". In: *SIAM Journal on Discrete Mathematics* 11.4 (Nov. 1998), pp. 603–623. DOI: 10.1137/S0895480197318878.

[BSS22] Jørgen Bang-Jensen, Thomas Schweser, and Michael Stiebitz. "Digraphs and Variable Degeneracy". In: *SIAM Journal on Discrete Mathematics* 36.1 (Mar. 2022), pp. 578–595. DOI: 10.1137/20M1386827.

[BSH22] Jørgen Bang-Jensen, Jonas Costa Ferreira da Silva, and Frédéric Havet. "On the Inversion Number of Oriented Graphs". In: *Discrete Mathematics & Theoretical Computer Science* vol. 23 no. 2, special issue in honour of Maurice Pouzet.Special issues (Dec. 2022). DOI: 10.46298/dmtcs.7474.

[BG96] Ann Becker and Dan Geiger. "Optimization of Pearl's Method of Conditioning and Greedy-like Approximation Algorithms for the Vertex Feedback Set Problem". In: *Artificial Intelligence* 83.1 (May 1996), pp. 167–188. DOI: 10.1016/0004-3702(95)00004-6.

[Bel+10] Houmem Belkhechine, Moncef Bouaziz, Imed Boudabbous, and Maurice Pouzet. "Inversion Dans Les Tournois". In: *Comptes Rendus Mathematique* 348.13 (July 2010), pp. 703–707. DOI: 10.1016/j.crma.2010.06.022.

[BM22] Alistair Benford and Richard Montgomery. "Trees with Few Leaves in Tournaments". In: *Journal of Combinatorial Theory, Series B* 155 (July 2022), pp. 141–170. DOI: 10.1016/j.jctb.2022.02.005.

[Ber61] Claude Berge. "Farbung von Graphen, Deren Samtliche Bzw. Deren Ungerade Kreise Starr Sind". In: *Wissenschaftliche Zeitschrift* (1961).

[Ber+13] Eli Berger, Krzysztof Choromanski, Maria Chudnovsky, Jacob Fox, Martin Loebl, Alex Scott, Paul Seymour, and Stéphan Thomassé. "Tournaments and Colouring". In: *Journal of Combinatorial Theory, Series B* 103.1 (Jan. 2013), pp. 1–20. DOI: 10.1016/j.jctb.2012.08.003.

[BT81] J. C. Bermond and C. Thomassen. "Cycles in Digraphs– a Survey". In: *Journal of Graph Theory* 5.1 (1981), pp. 1–43. DOI: 10.1002/jgt.3190050102.

[Bes+11] Stéphane Bessy, Fedor V. Fomin, Serge Gaspers, Christophe Paul, Anthony Perez, Saket Saurabh, and Stéphan Thomassé. "Kernels for Feedback Arc Set in Tournaments". In: *Journal of Computer and System Sciences* 77.6 (Nov. 2011), pp. 1071–1078. DOI: 10.1016/j.jcss.2010.10.001.

[BGR24] Stéphane Bessy, Daniel Gonçalves, and Amadeus Reinald. "Oriented Trees in O(K√k)-Chromatic Digraphs, a Subquadratic Bound for Burr's Conjecture". In: *Graph-Theoretic Concepts in Computer Science: 50th International Workshop, WG 2024, Gozd Martuljek, Slovenia, June 19–21, 2024, Revised Selected Papers*. Berlin, Heidelberg: Springer-Verlag, June 2024, pp. 79–91. DOI: 10.1007/978-3-031-75409-8_6.

[Bes+25] Stéphane Bessy, Daniel Gonçalves, Amadeus Reinald, and Dimitrios M. Thilikos. *Plane Strong Connectivity Augmentation*. Dec. 2025. DOI: 10.48550/arXiv.2512.17904. arXiv: 2512.17904 [math].

[BLW86] Norman Biggs, E Keith Lloyd, and Robin J Wilson. *Graph Theory, 1736-1936*. Oxford University Press, 1986.

[Bod94] Hans L. Bodlaender. "On Disjoint Cycles". In: *International Journal of Foundations of Computer Science* 05.01 (Mar. 1994), pp. 59–68. DOI: 10.1142/S0129054194000049.

[Bok+04] Drago Bokal, Gasper Fijavz, Martin Juvan, P. Mark Kayll, and Bojan Mohar. "The Circular Chromatic Number of a Digraph". In: *Journal of Graph Theory* 46.3 (2004), pp. 227–240. DOI: 10.1002/jgt.20003.

[BT98] B. Bollobás and A. Thomason. "Proof of a Conjecture of Mader, Erdös and Hajnal on Topological Complete Subgraphs". In: *European Journal of Combinatorics* 19.8 (Nov. 1998), pp. 883–887. DOI: 10.1006/eujc.1997.0188.

[BM79] Béla Bollobás and Bennet Manvel. "Optimal Vertex Partitions". In: *Bulletin of the London Mathematical Society* 11.2 (1979), pp. 113–116. DOI: 10.1112/blms/11.2.113.

[BG25] Marthe Bonamy and Colin Geniet. *χ-Boundedness and Neighbourhood Complexity of Bounded Merge-Width Graphs*. Apr. 2025. DOI: 10.48550/arXiv.2504.08266. arXiv: 2504.08266 [math].

[BP20] Marthe Bonamy and Michał Pilipczuk. "Graphs of Bounded Cliquewidth Are Polynomially Chi-Bounded". In: *Advances in Combinatorics* (July 2020). DOI: 10.19086/aic.13668.

[Bon76] J. A. Bondy. "Diconnected Orientations and a Conjecture of Las Vergnas". In: *Journal of the London Mathematical Society* s2-14.2 (1976), pp. 277–282. DOI: 10.1112/jlms/s2-14.2.277.

[Bon+23] Édouard Bonnet, Romain Bourneuf, Julien Duron, Colin Geniet, Stéphan Thomassé, and Nicolas Trotignon. *A Tamed Family of Triangle-Free Graphs with Unbounded Chromatic Number*. Apr. 2023. DOI: 10.48550/arXiv.2304.04296. arXiv: 2304.04296 [math].

[Bor79] O. V. Borodin. "Problems of Colouring and of Covering the Vertex Set of a Graph by Induced Subgraphs". In: *PhD Thesis* (1979).

[BKT00] O. V. Borodin, A. V. Kostochka, and B. Toft. "Variable Degeneracy: Extensions of Brooks' and Gallai's Theorems". In: *Discrete Mathematics* 214.1 (Mar. 2000), pp. 101–112. DOI: 10.1016/S0012-365X(99)00221-6.

[Bor76] Oleg V. Borodin. "On Decomposition of Graphs into Degenerate Subgraphs". In: *Diskretny analys, Novosibirsk* 28 (1976), pp. 3–12.

[BKM09] Glencora Borradaile, Philip Klein, and Claire Mathieu. "An O(n Log n) Approximation Scheme for Steiner Tree in Planar Graphs". In: *ACM Trans. Algorithms* 5.3 (July 2009), 31:1–31:31. DOI: 10.1145/1541885.1541892.

[BT25] Romain Bourneuf and Stéphan Thomassé. "Bounded Twin-Width Graphs Are Polynomially χ-Bounded". In: *Advances in Combinatorics* (Feb. 2025). DOI: 10.19086/aic.2025.2.

[Bou+24] N. Bousquet, F. Havet, N. Nisse, L. Picasarri-Arrieta, and A. Reinald. "Digraph Redicolouring". In: *European Journal of Combinatorics* 116 (Feb. 2024), p. 103876. DOI: 10.1016/j.ejc.2023.103876.

[BDW23] Marcin Briański, James Davies, and Bartosz Walczak. "Separating Polynomial χ-Boundedness from χ-Boundedness". In: *Combinatorica* (Aug. 2023). DOI: 10.1007/s00493-023-00054-3. arXiv: 2201.08814 [cs, math].

[Bro41] R. L. Brooks. "On Colouring the Nodes of a Network". In: *Mathematical Proceedings of the Cambridge Philosophical Society* 37.2 (Apr. 1941), pp. 194–197. DOI: 10.1017/S030500410002168X.

[Buc18] Matija Bucić. "An Improved Bound for Disjoint Directed Cycles". In: *Discrete Mathematics* 341.8 (Aug. 2018), pp. 2231–2236. DOI: 10.1016/j.disc.2018.04.027.

[Bur65] James Perkins Burling. "On Coloring Problems of Families of Polytopes." PhD thesis. Jan. 1965.

[Bur80] Stefan A. Burr. "Subtrees of Directed Graphs and Hypergraphs". In: *Proceedings of the Eleventh Southeastern Conference on Combinatorics, Graph Theory and Computing, Boca Raton, Congr. Numer* 28 (1980), pp. 227–239.

[CH78] Louis Caccetta and Roland Haggkvist. *On Minimal Digraphs with given Girth*. Utilitas Mathematics. Department of Combinatorics and Optimization, University of Waterloo, 1978.

[CS89] Guo-Ray Cai and Yu-Geng Sun. "The Minimum Augmentation of Any Graph to a K-edge-connected Graph". In: *Networks. An International Journal* 19.1 (1989), pp. 151–172. DOI: 10.1002/net.3230190112.

[CE91] Leizhen Cai and John A. Ellis. "NP-completeness of Edge-Colouring Some Restricted Graphs". In: *Discrete Applied Mathematics* 30.1 (Jan. 1991), pp. 15–27. DOI: 10.1016/0166-218X(91)90010-T.

[Cam59] Paul Camion. "Chemins et Circuits Hamiltoniens Des Graphes Complets". In: *Comptes Rendus Hebdomadaires Des Seances De L Academie Des Sciences* 249.21 (1959), pp. 2151–2152.

[Cam+] Marcelo Campos, Simon Griffiths, Robert Morris, and Julian Sahasrabudhe. "An Exponential Improvement for Diagonal Ramsey | Annals of Mathematics". In: *Annals of Mathematics* (to appear) ().

[Car+23] Alvaro Carbonero, Patrick Hompe, Benjamin Moore, and Sophie Spirkl. "A Counterexample to a Conjecture about Triangle-Free Induced Subgraphs of Graphs with Large Chromatic Number". In: *Journal of Combinatorial Theory, Series B* 158 (Jan. 2023), pp. 63–69. DOI: 10.1016/j.jctb.2022.09.001.

[Car+22] Alvaro Carbonero, Patrick Hompe, Benjamin Moore, and Sophie Spirkl. "Digraphs with All Induced Directed Cycles of the Same Length Are Not χ-Bounded". In: *The Electronic Journal of Combinatorics* (Oct. 2022), P4.4–P4.4. DOI: 10.37236/11179.

[Car+25] Alvaro Carbonero, Hidde Koerts, Benjamin Moore, and Sophie Spirkl. “On Heroes in Digraphs with Forbidden Induced Forests”. In: *European Journal of Combinatorics* 125 (Mar. 2025), p. 104104. DOI: 10.1016/j.ejc.2024.104104.

[Cat79] Paul A Catlin. “Hajós’ Graph-Coloring Conjecture: Variations and Counterexamples”. In: *Journal of Combinatorial Theory, Series B* 26.2 (Apr. 1979), pp. 268–274. DOI: 10.1016/0095-8956(79)90062-5.

[CTY07] Pierre Charbit, Stéphan Thomassé, and Anders Yeo. “The Minimum Feedback Arc Set Problem Is NP-Hard for Tournaments”. In: *Combinatorics, Probability and Computing* 16.01 (Jan. 2007), p. 1. DOI: 10.1017/S0963548306007887.

[Che+08] Jianer Chen, Yang Liu, Songjian Lu, Barry O’sullivan, and Igor Razgon. “A Fixed-Parameter Algorithm for the Directed Feedback Vertex Set Problem”. In: *J. ACM* 55.5 (Nov. 2008), 21:1–21:19. DOI: 10.1145/1411509.1411511.

[Chi+20] Rajesh H. Chitnis, Andreas E. Feldmann, MohammadTaghi HajiAghayi, and Daniel Marx. “Tight Bounds for Planar Strongly Connected Steiner Subgraph with Fixed Number of Terminals (and Extensions)”. In: *SIAM Journal on Computing* 49.2 (Jan. 2020), pp. 318–364. DOI: 10.1137/18M122371X.

[CS25] Micha Christoph and Raphael Steiner. *Proof of the KAMAK Tree Conjecture*. May 2025. DOI: 10.48550/arXiv.2505.21367. arXiv: 2505.21367 [math].

[Chu+06] Maria Chudnovsky, Neil Robertson, Paul Seymour, and Robin Thomas. “The Strong Perfect Graph Theorem”. In: (2006). DOI: 10.4007/annals.2006.164.51.

[CSS19] Maria Chudnovsky, Alex Scott, and Paul Seymour. “Induced Subgraphs of Graphs with Large Chromatic Number. XI. Orientations”. In: *European Journal of Combinatorics* 76 (Feb. 2019), pp. 53–61. DOI: 10.1016/j.ejc.2018.09.003.

[CS14] Maria Chudnovsky and Paul Seymour. “Extending the Gyárfás–Sumner Conjecture”. In: *Journal of Combinatorial Theory, Series B* 105 (Mar. 2014), pp. 11–16. DOI: 10.1016/j.jctb.2013.11.002.

[Chu81] F. R. K. Chung. *A Note on Subtrees in Tournaments*. 1981.

[CL74] V. Chvátal and L. Lovász. “Every Directed Graph Has a Semi-Kernel”. In: *Hypergraph Seminar*. Ed. by Claude Berge and Dijen Ray-Chaudhuri. Lecture Notes in Mathematics. Berlin, Heidelberg: Springer, 1974, pp. 175–175. DOI: 10.1007/BFb0066192.

[Coh+18a] Nathann Cohen, Frédéric Havet, William Lochet, and Raul Lopes. “Bispindles in Strongly Connected Digraphs with Large Chromatic Number”. In: *The Electronic Journal of Combinatorics* (June 2018), P2.39–P2.39. DOI: 10.37236/6922.

[Coh+18b] Nathann Cohen, Frédéric Havet, William Lochet, and Nicolas Nisse. “Subdivisions of Oriented Cycles in Digraphs with Large Chromatic Number”. In: *Journal of Graph Theory* 89.4 (Dec. 2018), pp. 439–456. DOI: 10.1002/jgt.22360.

[Coo+23] Linda Cook, Tomáš Masařík, Marcin Pilipczuk, Amadeus Reinald, and Uéverton S. Souza. “Proving a Directed Analogue of the Gyárfás-Sumner Conjecture for Orientations of P4”. In: *The Electronic Journal of Combinatorics* (Sept. 2023), P3.36–P3.36. DOI: 10.37236/11538.

[Cor+23] Timothée Corsini, Quentin Deschamps, Carl Feghali, Daniel Gonçalves, Hélène Langlois, and Alexandre Talon. "Partitioning into Degenerate Graphs in Linear Time". In: *European Journal of Combinatorics* 114 (Dec. 2023), p. 103771. DOI: 10.1016/j.ejc.2023.103771.

[Cou90] Bruno Courcelle. "The Monadic Second-Order Logic of Graphs. I. Recognizable Sets of Finite Graphs". In: *Information and Computation* 85.1 (Mar. 1990), pp. 12–75. DOI: 10.1016/0890-5401(90)90043-H.

[Cyg+15] Marek Cygan, Fedor V. Fomin, Łukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Cham: Springer International Publishing, 2015. DOI: 10.1007/978-3-319-21275-3.

[DP25] Michelle Delcourt and Luke Postle. "Reducing Linear Hadwiger's Conjecture to Coloring Small Graphs". In: *Journal of the American Mathematical Society* 38.2 (Apr. 2025), pp. 481–507. DOI: 10.1090/jams/1047.

[Des54] Blanche Descartes. "Solution to Advanced Problem No. 4526". In: *The American Mathematical Monthly* 61.352 (1954), p. 216.

[DL10] Emilio Di Giacomo and Giuseppe Liotta. "The Hamiltonian Augmentation Problem and Its Applications to Graph Drawing". In: *WALCOM: Algorithms and Computation*. Ed. by Md. Saidur Rahman and Satoshi Fujita. Berlin, Heidelberg: Springer, 2010, pp. 35–46. DOI: 10.1007/978-3-642-11440-3_4.

[Die17] Reinhard Diestel. *Graph Theory*. Vol. 173. Graduate Texts in Mathematics. Berlin, Heidelberg: Springer, 2017. DOI: 10.1007/978-3-662-53622-3.

[DF99] R. G. Downey and M. R. Fellows. *Parameterized Complexity*. Ed. by David Gries and Fred B. Schneider. Monographs in Computer Science. New York, NY: Springer, 1999. DOI: 10.1007/978-1-4612-0515-9.

[DW71] S. E. Dreyfus and R. A. Wagner. "The Steiner Problem in Graphs". In: *Networks* 1.3 (1971), pp. 195–207. DOI: 10.1002/net.3230010302.

[DH21] François Dross and Frédéric Havet. "On the Unavoidability of Oriented Trees". In: *Journal of Combinatorial Theory, Series B* 151 (Nov. 2021), pp. 83–110. DOI: 10.1016/j.jctb.2021.06.003.

[Dur+25] Lech Duraj, Ross J. Kang, Hoang La, Jonathan Narboni, Filip Pokrývka, Clément Rambaud, and Amadeus Reinald. "The χ-Binding Function of d-Directional Segment Graphs". In: *Discrete & Computational Geometry* (May 2025). DOI: 10.1007/s00454-025-00737-2.

[Dur+24] Julien Duron, Frédéric Havet, Florian Hörsch, and Clément Rambaud. *On the Minimum Number of Inversions to Make a Digraph K-(Arc-)Strong*. May 2024. DOI: 10.48550/arXiv.2303.11719. arXiv: 2303.11719 [math].

[EG77] Jack Edmonds and Rick Giles. "A Min-Max Relation for Submodular Functions on Graphs". In: *Annals of Discrete Mathematics*. Ed. by P. L. Hammer, E. L. Johnson, B. H. Korte, and G. L. Nemhauser. Vol. 1. Studies in Integer Programming. Elsevier, Jan. 1977, pp. 185–204. DOI: 10.1016/S0167-5060(08)70734-9.

[El 22] Mouhamad El Joubbeh. "On Three Blocks Paths P(k,l,r)". In: *Discrete Applied Mathematics* 322 (Dec. 2022), pp. 237–239. DOI: 10.1016/j.dam.2022.08.021.

[EG24] Mouhamad El Joubbeh and Salman Ghazal. "Existence of Paths with t Blocks in k (t) -Chromatic Digraph". In: *Discrete Applied Mathematics* 342 (Jan. 2024), pp. 381–384. DOI: 10.1016/j.dam.2023.09.025.

[El 04a] A. El Sahili. "Trees in Tournaments". In: *Journal of Combinatorial Theory, Series B* 92.1 (Sept. 2004), pp. 183–187. DOI: 10.1016/j.jctb.2004.04.002.

[El 04b] Amine El Sahili. "Paths with Two Blocks in K-Chromatic Digraphs". In: *Discrete Mathematics* 287.1 (Oct. 2004), pp. 151–153. DOI: 10.1016/j.disc.2004.03.013.

[EG59] P Erdős and T. Gallai. "On Maximal Paths and Circuits of Graphs". In: *Acta Mathematica Academiae Scientiarum Hungarica* 10.3 (Sept. 1959), pp. 337–356. DOI: 10.1007/BF02024498.

[EP65] p. Erdös and L. Pósa. "On Independent Circuits Contained in a Graph". In: *Canadian Journal of Mathematics* 17 (1965), pp. 347–352. DOI: 10.4153/CJM-1965-035-8.

[Erd79a] Paul Erdös. "Choosability in Graphs". In: *Congressus Numerantium* 26.4 (1979), pp. 125–157.

[Erd59] Paul Erdös. "Graph Theory and Probability". In: *Canadian Journal of Mathematics* 11 (Jan. 1959), pp. 34–38. DOI: 10.4153/CJM-1959-003-9.

[Erd88] Paul Erdös. "Problems and Results in Combinatorial Analysis and Graph Theory". In: *Annals of Discrete Mathematics*. Ed. by J. Akiyama, Y. Egawa, and H. Enomoto. Vol. 38. Graph Theory and Applications. Elsevier, Jan. 1988, pp. 81–92. DOI: 10.1016/S0167-5060(08)70773-8.

[Erd79b] Paul Erdös. "Problems and Results in Number Theory". In: *Proc. Ninth Manitoba Conference on Numerical Math. and Computing* (1979), pp. 3–21.

[EH89] Paul Erdös and A. Hajnal. "Ramsey-Type Theorems". In: *Discrete Applied Mathematics* 25.1 (Oct. 1989), pp. 37–52. DOI: 10.1016/0166-218X(89)90045-0.

[EM65] Paul Erdös and J. W. Moon. "On Sets of Consistent Arcs in a Tournament". In: *Canadian Mathematical Bulletin* 8.3 (Apr. 1965), pp. 269–271. DOI: 10.4153/CMB-1965-017-1.

[Esp17] Louis Esperet. "Graph Colorings, Flows and Perfect Matchings". Accreditation to Supervise Research. Université Grenoble Alpes, Oct. 2017.

[ET76] Kapali P. Eswaran and R. Endre Tarjan. "Augmentation Problems". In: *SIAM Journal on Computing* 5.4 (Dec. 1976), pp. 653–665. DOI: 10.1137/0205044.

[Eul41] Leonhard Euler. "Solutio Problematis Ad Geometriam Situs Pertinentis". In: *Commentarii academiae scientiarum Petropolitanae* (Jan. 1741), pp. 128–140.

[Fel+09] Michael R. Fellows, Danny Hermelin, Frances Rosamond, and Stéphane Vialette. "On the Parameterized Complexity of Multiple-Interval Graph Problems". In: *Theoretical Computer Science* 410.1 (Jan. 2009), pp. 53–61. DOI: 10.1016/j.tcs.2008.09.065.

[Fer83] W Fernandez de la Vega. "On the Maximum Cardinality of a Consistent Set of Arcs in a Random Tournament". In: *Journal of Combinatorial Theory, Series B* 35.3 (Dec. 1983), pp. 328–332. DOI: 10.1016/0095-8956(83)90060-6.

[FF56] L.R. Ford and D. R. Fulkerson. "Maximal Flow Through a Network". In: *Canadian Journal of Mathematics* 8 (Jan. 1956), pp. 399–404. DOI: 10.4153/CJM-1956-045-5.

[FJ95] A. Frank and T. Jordan. "Minimal Edge-Coverings of Pairs of Sets". In: *Journal of Combinatorial Theory, Series B* 65.1 (Sept. 1995), pp. 73–110. DOI: 10.1006/jctb.1995.1044.

[Fra82] András Frank. "An Algorithm for Submodular Functions on Graphs". In: *North-Holland Mathematics Studies*. Ed. by Achim Bachem, Martin Grötschel, and Bemhard Korte. Vol. 66. Bonn Workshop on Combinatorial Optimization. North-Holland, Jan. 1982, pp. 97–120. DOI: 10.1016/S0304-0208(08)72446-0.

[Fra92] András Frank. "Augmenting Graphs to Meet Edge-Connectivity Requirements". In: *SIAM Journal on Discrete Mathematics* 5.1 (Feb. 1992), pp. 25–53. DOI: 10.1137/0405003.

[Fra94] András Frank. "Connectivity Augmentation Problems in Network Design". In: *Mathematical Programming : State of the Art 1994* (1994).

[Fra84] András Frank. "Finding Feasible Vectors of Edmonds-Giles Polyhedra". In: *Journal of Combinatorial Theory, Series B* 36.3 (June 1984), pp. 221–239. DOI: 10.1016/0095-8956(84)90029-7.

[Fra81] András Frank. "How to Make a Digraph Strongly Connected". In: *Combinatorica* 1.2 (June 1981), pp. 145–153. DOI: 10.1007/BF02579270.

[FJ15] András Frank and Tibor Jordán. "Graph Connectivity Augmentation". In: *Handbook of Graph Theory, Combinatorial Optimization, and Algorithms* (2015), pp. 313–346.

[FC70] H. Frank and Wushow Chou. "Connectivity Considerations in the Design of Survivable Networks". In: *IEEE Transactions on Circuit Theory* 17.4 (Nov. 1970), pp. 486–490. DOI: 10.1109/TCT.1970.1083185.

[FJ81] Greg N. Frederickson and Joseph Ja'Ja'. "Approximation Algorithms for Several Graph Augmentation Problems". In: *SIAM Journal on Computing* 10.2 (May 1981), pp. 270–283. DOI: 10.1137/0210019.

[GJ00] Harold N. Gabow and Tibor Jordán. "How to Make a Square Grid Framework with Cables Rigid". In: *SIAM Journal on Computing* 30.2 (Jan. 2000), pp. 649–680. DOI: 10.1137/S0097539798347189.

[Gal+24] Esther Galby, Sándor Kisfaludi-Bak, Dániel Marx, and Roohani Sharma. "Subexponential Parameterized Directed Steiner Network Problems on Planar Graphs: A Complete Classification". In: *LIPIcs, Volume 297, ICALP 2024* 297 (2024). Ed. by Karl Bringmann, Martin Grohe, Gabriele Puppis, and Ola Svensson, 67:1–67:19. DOI: 10.4230/LIPICS.ICALP.2024.67.

[Gal68] T. Gallai. "On Directed Paths and Circuits". In: *Theory of graphs* 38 (1968), p. 2054.

[GJS76] M. R. Garey, D. S. Johnson, and L. Stockmeyer. "Some Simplified *NP*-Complete Graph Problems". In: *Theoretical Computer Science* 1.3 (Feb. 1976), pp. 237–267. DOI: 10.1016/0304-3975(76)90059-1.

[GJ09] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completeness*. 27. print. A Series of Books in the Mathematical Sciences. New York [u.a]: Freeman, 2009.

[GRT15] Petr A. Golovach, Clément Requilé, and Dimitrios M. Thilikos. "Variants of Plane Diameter Completion". In: *LIPIcs, Volume 43, IPEC 2015* 43 (2015), pp. 30–42. DOI: 10.4230/LIPICS.IPEC.2015.30.

[GPR25] Daniel Gonçalves, Lucas Picasarri-Arrieta, and Amadeus Reinald. "Brooks-Type Colourings of Digraphs in Linear Time". In: *Journal of Graph Theory* 110.4 (2025), pp. 496–513. DOI: 10.1002/jgt.23266.

[Gra70] R. L. Graham. "On Subtrees of Directed Graphs with No Path of Length Exceeding One". In: *Canadian Mathematical Bulletin* 13.3 (Sept. 1970), pp. 329–332. DOI: 10.4153/CMB-1970-063-1.

[GS71] R. L. Graham and J. H. Spencer. "A Constructive Solution to a Tournament Problem". In: *Canadian Mathematical Bulletin* 14.1 (Jan. 1971), pp. 45–48. DOI: 10.4153/CMB-1971-007-1.

[GM92] Stephen Guattery and Gary L. Miller. "A Contraction Procedure for Planar Directed Graphs". In: *Proceedings of the Fourth Annual ACM Symposium on Parallel Algorithms and Architectures*. San Diego California USA: ACM, June 1992, pp. 431–441. DOI: 10.1145/140901.141935.

[GNS11] Jiong Guo, Rolf Niedermeier, and Ondřej Suchý. "Parameterized Complexity of Arc-Weighted Directed Steiner Problems". In: *SIAM Journal on Discrete Mathematics* 25.2 (Jan. 2011), pp. 583–599. DOI: 10.1137/100794560.

[GL16] Venkatesan Guruswami and Euiwoong Lee. "Simple Proof of Hardness of Feedback Vertex Set". In: *Theory of Computing* 12.6 (Aug. 2016), pp. 1–11. DOI: 10.4086/toc.2016.v012a006.

[GMZ09] Carsten Gutwenger, Petra Mutzel, and Bernd Zey. "Planar Biconnectivity Augmentation with Fixed Embedding". In: *Combinatorial Algorithms*. Ed. by Jiří Fiala, Jan Kratochvíl, and Mirka Miller. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2009, pp. 289–300. DOI: 10.1007/978-3-642-10217-2_29.

[GST80] A. Gyárfás, E. Szemeredi, and Zs. Tuza. "Induced Subtrees in Graphs of Large Chromatic Number". In: *Discrete Mathematics* 30.3 (Jan. 1980), pp. 235–244. DOI: 10.1016/0012-365X(80)90230-7.

[Gyá89] András Gyárfás. "Problem 115". In: Discrete Mathematics.79 (1989), pp. 109–110.

[Gyá87] András Gyárfás. "Problems from the World Surrounding Perfect Graphs". In: *Applicationes Mathematicae* 19.3-4 (1987), pp. 413–441. DOI: 10.4064/am-19-3-4-413-441.

[Har+17] Ararat Harutyunyan, Tien-Nam Le, Alantha Newman, and Stéphan Thomassé. "Coloring Dense Digraphs". In: *Electronic Notes in Discrete Mathematics*. The European Conference on Combinatorics, Graph Theory and Applications (EUROCOMB'17) 61 (Aug. 2017), pp. 577–583. DOI: 10.1016/j.endm.2017.07.010.

[HM11] Ararat Harutyunyan and Bojan Mohar. "Gallai's Theorem for List Coloring of Digraphs". In: *SIAM Journal on Discrete Mathematics* 25.1 (Jan. 2011), pp. 170–180. DOI: 10.1137/100803870.

[HM12] Ararat Harutyunyan and Bojan Mohar. "Two Results on the Digraph Chromatic Number". In: *Discrete Mathematics* 312.10 (May 2012), pp. 1823–1826. DOI: 10.1016/j.disc.2012.01.028.

[Has65] Maria Hasse. "Zur Algebraischen Begründung Der Graphentheorie. I". In: *Mathematische Nachrichten* 28.5-6 (1965), pp. 275–290. DOI: 10.1002/mana.19650280503.

[Hav03] F. Havet. "On Unavoidability of Trees with k Leaves". In: *Graphs and Combinatorics* 19.1 (Mar. 2003), pp. 101–110. DOI: 10.1007/s00373-002-0483-y.

[Hav00] Frédéric Havet. "Oriented Hamiltonian Cycles in Tournaments". In: *Journal of Combinatorial Theory, Series B* 80.1 (Sept. 2000), pp. 1–31. DOI: 10.1006/jctb.2000.1959.

[HHR24] Frédéric Havet, Florian Hörsch, and Clément Rambaud. *Diameter of the Inversion Graph.* May 2024. DOI: 10.48550/arXiv.2405.04119. arXiv: 2405.04119 [math].

[HT00a] Frédéric Havet and Stéphan Thomassé. "Median Orders of Tournaments: A Tool for the Second Neighborhood Problem and Sumner's Conjecture". In: *Journal of Graph Theory* 35.4 (Dec. 2000), pp. 244–256. DOI: 10.1002/1097-0118(200012)35:4<244::AID-JGT2>3.0.CO;2-H.

[HT00b] Frédéric Havet and Stéphan Thomassé. "Oriented Hamiltonian Paths in Tournaments: A Proof of Rosenfeld's Conjecture". In: *Journal of Combinatorial Theory, Series B* 78.2 (Mar. 2000), pp. 243–273. DOI: 10.1006/jctb.1999.1945.

[HKN17] Jan Hladky, Daniel Kral, and Sergey Norin. "Counting Flags in Triangle-Free Digraphs". In: *Combinatorica* 37.1 (Feb. 2017), pp. 49–76. DOI: 10.1007/s00493-015-2662-5. arXiv: 0908.2791 [math].

[Hon+25] Tomáš Hons, Tereza Klimošová, Gaurav Kucheriya, David Mikšaník, Josef Tkadlec, and Mykhaylo Tyomkyn. *Unavoidable Subgraphs in Digraphs with Large Out-Degrees.* Apr. 2025. DOI: 10.48550/arXiv.2504.20616. arXiv: 2504.20616 [math].

[HT13] Ferran Hurtado and Csaba D. Tóth. "Plane Geometric Graph Augmentation: A Generic Perspective". In: *Thirty Essays on Geometric Graph Theory.* Ed. by János Pach. New York, NY: Springer New York, 2013, pp. 327–354. DOI: 10.1007/978-1-4614-0110-0_17.

[Jac81] Bill Jackson. "Long Paths and Cycles in Oriented Graphs". In: *Journal of Graph Theory* 5.2 (June 1981), pp. 145–157. DOI: 10.1002/jgt.3190050204.

[JJ05] Bill Jackson and Tibor Jordán. "Independence Free Graphs and Vertex Connectivity Augmentation". In: *Journal of Combinatorial Theory, Series B* 94.1 (May 2005), pp. 31–77. DOI: 10.1016/j.jctb.2004.01.004.

[Jar30] Vojtěch Jarník. "O jistém problému minimálním. (Z dopisu panu O. Borůvkovi)". In: (1930).

[Joh+01] Thor Johnson, Neil Robertson, P.D. Seymour, and Robin Thomas. "Directed Tree-Width". In: *Journal of Combinatorial Theory, Series B* 82.1 (May 2001), pp. 138–154. DOI: 10.1006/jctb.2000.2031.

[Jor95] T. Jordan. "On the Optimal Vertex-Connectivity Augmentation". In: *Journal of Combinatorial Theory, Series B* 63.1 (Jan. 1995), pp. 8–20. DOI: 10.1006/jctb.1995.1002.

[Jor97] Tibor Jordan. *Two NP-Complete Augmentation Problems.* Technical Report. University of Southern Denmark, Feb. 1997.

[Kan96] Goos Kant. "Augmenting Outerplanar Graphs". In: *Journal of Algorithms* 21.1 (July 1996), pp. 1–25. DOI: 10.1006/jagm.1996.0034.

[KB91] Goos Kant and Hans L. Bodlaender. “Planar Graph Augmentation Problems”. In: *Algorithms and Data Structures.* Ed. by Frank Dehne, Jörg-Rüdiger Sack, and Nicola Santoro. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 1991, pp. 286–298. DOI: 10.1007/BFb0028270.

[Kar72] Richard M. Karp. “Reducibility among Combinatorial Problems”. In: *Complexity of Computer Computations: Proceedings of a Symposium on the Complexity of Computer Computations, Held March 20–22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York, and Sponsored by the Office of Naval Research, Mathematics Program, IBM World Trade Corporation, and the IBM Research Mathematical Sciences Department.* Ed. by Raymond E. Miller, James W. Thatcher, and Jean D. Bohlinger. Boston, MA: Springer US, 1972, pp. 85–103. DOI: 10.1007/978-1-4684-2001-2_9.

[KK15] Ken-ichi Kawarabayashi and Stephan Kreutzer. “The Directed Grid Theorem”. In: *Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing.* STOC ’15. New York, NY, USA: Association for Computing Machinery, June 2015, pp. 655–664. DOI: 10.1145/2746539.2746586.

[KP25] Ken-ichi Kawarabayashi and Lucas Picasarri-Arrieta. “An Analogue of Reed’s Conjecture for Digraphs”. In: *Proceedings of the 2025 annual ACM-SIAM symposium on discrete algorithms (SODA)* (2025), pp. 3310–3324. DOI: 10.1137/1.9781611978322.106. eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9781611978322.106.

[KS07] Claire Kenyon-Mathieu and Warren Schudy. “How to Rank with Few Errors”. In: *Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing.* San Diego California USA: ACM, June 2007, pp. 95–103. DOI: 10.1145/1250790.1250806.

[KT92] H. A. Kierstead and W. T. Trotter. “Colorful Induced Subgraphs”. In: *Discrete Mathematics* 101.1 (May 1992), pp. 165–169. DOI: 10.1016/0012-365X(92)90600-K.

[KS15] Ilhee Kim and Paul Seymour. “Tournament Minors”. In: *Journal of Combinatorial Theory, Series B* 112 (May 2015), pp. 138–153. DOI: 10.1016/j.jctb.2014.12.005.

[Kim+18] Ringi Kim, Seog-Jin Kim, Jie Ma, and Boram Park. “Cycles with Two Blocks in K-Chromatic Digraphs”. In: *Journal of Graph Theory* 88.4 (2018), pp. 592–605. DOI: 10.1002/jgt.22232.

[KMS21] Kristine Vitting Klinkby, Pranabendu Misra, and Saket Saurabh. “Strong Connectivity Augmentation Is FPT”. In: *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA).* Proceedings. Society for Industrial and Applied Mathematics, Jan. 2021, pp. 219–234. DOI: 10.1137/1.9781611976465.15.

[KP14] Tomasz Kociumaka and Marcin Pilipczuk. “Faster Deterministic Feedback Vertex Set”. In: *Information Processing Letters* 114.10 (Oct. 2014), pp. 556–560. DOI: 10.1016/j.ipl.2014.05.001.

[KS96] János Komlós and Endre Szemerédi. “Topological Cliques in Graphs II”. In: *Combinatorics, Probability and Computing* 5.1 (Mar. 1996), pp. 79–90. DOI: 10.1017/S096354830000184X.

[KPS24] Tuukka Korhonen, Michał Pilipczuk, and Giannos Stamoulis. "Minor Containment and Disjoint Paths in Almost-Linear Time". In: *2024 IEEE 65th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE Computer Society, Oct. 2024, pp. 53–61. DOI: `10.1109/FOCS61266.2024.00014`.

[KSS23] Alexandr V. Kostochka, Thomas Schweser, and Michael Stiebitz. "Generalized DP-colorings of Graphs". In: *Discrete Mathematics*. Special Issue in Honour of Landon Rabern 346.11 (Nov. 2023), p. 113186. DOI: `10.1016/j.disc.2022.113186`.

[KST54] T. Kóvari, V. Sós, and P. Turán. "On a Problem of K. Zarankiewicz". In: *Colloquium Mathematicae* 3.1 (1954), pp. 50–57. DOI: `10.4064/cm-3-1-50-57`.

[KMO11] Daniela Kühn, Richard Mycroft, and Deryk Osthus. "A Proof of Sumner's Universal Tournament Conjecture for Large Tournaments". In: *Proceedings of the London Mathematical Society* 102.4 (2011), pp. 731–766. DOI: `10.1112/plms/pdq035`.

[KO04] Daniela Kühn and Deryk Osthus. "Induced Subdivisions In Ks,s-Free Graphs of Large Average Degree". In: *Combinatorica* 24.2 (Apr. 2004), pp. 287–304. DOI: `10.1007/s00493-004-0017-8`.

[Kur30] Casimir Kuratowski. "Sur le problème des courbes gauches en Topologie". In: *Fundamenta Mathematicae* 15.1 (1930), pp. 271–283. DOI: `10.4064/fm-15-1-271-283`.

[LM17] Zhentao Li and Bojan Mohar. "Planar Digraphs of Digirth Four Are 2-Colorable". In: *SIAM Journal on Discrete Mathematics* 31.3 (Jan. 2017), pp. 2201–2205. DOI: `10.1137/16M108080X`.

[LPS09] Nicolas Lichiardopol, Attila Pór, and Jean-Sébastien Sereni. "A Step toward the Bermond–Thomassen Conjecture about Disjoint Cycles in Digraphs". In: *SIAM Journal on Discrete Mathematics* 23.2 (Jan. 2009), pp. 979–992. DOI: `10.1137/080715792`.

[Lic82] David Lichtenstein. "Planar Formulae and Their Uses". In: *SIAM Journal on Computing* 11.2 (1982), pp. 329–343. DOI: `10.1137/0211025`. eprint: `https://doi.org/10.1137/0211025`.

[Loc19] William Lochet. "Immersion of Transitive Tournaments in Digraphs with Large Minimum Outdegree". In: *Journal of Combinatorial Theory, Series B* 134 (Jan. 2019), pp. 350–353. DOI: `10.1016/j.jctb.2018.05.004`.

[Loc18] William Lochet. "Sous-structures dans les graphes dirige´s". In: (2018).

[Lov75] L Lovász. "Three Short Proofs in Graph Theory". In: *Journal of Combinatorial Theory, Series B* 19.3 (Dec. 1975), pp. 269–271. DOI: `10.1016/0095-8956(75)90089-1`.

[LY78] C. L. Lucchesi and D. H. Younger. "A Minimax Theorem for Directed Graphs". In: *Journal of the London Mathematical Society* s2-17.3 (1978), pp. 369–374. DOI: `10.1112/jlms/s2-17.3.369`.

[Mad85] W. Mader. "Degree and Local Connectivity in Digraphs". In: *Combinatorica* 5.2 (June 1985), pp. 161–165. DOI: `10.1007/BF02579379`.

[Mad72] W. Mader. "Existenzn-fach zusammenhängender Teilgraphen in Graphen genügend großer Kantendichte". In: *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg* 37.1 (Mar. 1972), pp. 86–97. DOI: `10.1007/BF02993903`.

[MPP18] Dániel Marx, Marcin Pilipczuk, and Michał Pilipczuk. "On Subexponential Parameterized Algorithms for Steiner Tree and Directed Subset TSP on Planar Graphs". In: *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*. Oct. 2018, pp. 474–484. DOI: 10.1109/FOCS.2018.00052.

[MV15] Dániel Marx and László A. Végh. "Fixed-Parameter Algorithms for Minimum-Cost Edge-Connectivity Augmentation". In: *ACM Trans. Algorithms* 11.4 (Apr. 2015), 27:1–27:24. DOI: 10.1145/2700210.

[Mas+05] T. Mashima, T. Fukuoka, S. Taoka, and T. Watanabe. "Minimum Augmentation to Bi-Connect Specified Vertices of a Graph with Upper Bounds on Vertex-Degree". In: *2005 IEEE International Symposium on Circuits and Systems (ISCAS)*. 2005, 752–755 Vol. 1. DOI: 10.1109/ISCAS.2005.1464697.

[Men27] Karl Menger. "Zur allgemeinen Kurventheorie". In: *Fundamenta Mathematicae* 10 (1927), pp. 96–115. DOI: 10.4064/fm-10-1-96-115.

[Myc55] Jan Mycielski. "Sur le coloriage des graphs". In: *Colloquium Mathematicae* 3.2 (1955), pp. 161–162. DOI: 10.4064/cm-3-2-161-162.

[MW18] Wendy Myrvold and Jennifer Woodcock. "A Large Set of Torus Obstructions and How They Were Discovered". In: *The Electronic Journal of Combinatorics* (Jan. 2018), P1.16–P1.16. DOI: 10.37236/3797.

[NE98] Hiroshi Nagamochi and Peter Eades. "Edge-Splitting and Edge-Connectivity Augmentation in Planar Graphs". In: *Integer Programming and Combinatorial Optimization*. Ed. by Robert E. Bixby, E. Andrew Boyd, and Roger Z. Ríos-Mercado. Berlin, Heidelberg: Springer, 1998, pp. 96–111. DOI: 10.1007/3-540-69346-7_8.

[Nas60] C. St J. A. Nash-Williams. "On Orientations, Connectivity and Odd-Vertex-Pairings in Finite Graphs". In: *Canadian Journal of Mathematics* 12 (Jan. 1960), pp. 555–567. DOI: 10.4153/CJM-1960-049-6.

[Neu82] V Neumann-Lara. "The Dichromatic Number of a Digraph". In: *Journal of Combinatorial Theory, Series B* 33.3 (Dec. 1982), pp. 265–270. DOI: 10.1016/0095-8956(82)90046-6.

[Paw+14] Arkadiusz Pawlik, Jakub Kozik, Tomasz Krawczyk, Michał Lasoń, Piotr Micek, William T. Trotter, and Bartosz Walczak. "Triangle-Free Intersection Graphs of Line Segments with Large Chromatic Number". In: *Journal of Combinatorial Theory, Series B* 105 (Mar. 2014), pp. 6–10. DOI: 10.1016/j.jctb.2013.11.001. arXiv: 1209.1595 [cs, math].

[Pen+25] Irena Penev, S. Taruni, Stéphan Thomassé, Ana Trujillo-Negrete, and Mykhaylo Tyomkyn. *Two-Block Paths in Oriented Graphs of Large Semidegree*. Mar. 2025. DOI: 10.48550/arXiv.2503.23191. arXiv: 2503.23191 [math].

[Pic24a] Lucas Picasarri-Arrieta. "Digraph Colouring". PhD thesis. Université Côte d'Azur, June 2024.

[Pic24b] Lucas Picasarri-Arrieta. "Strengthening the Directed Brooks' Theorem for Oriented Graphs and Consequences on Digraph Redicolouring". In: *Journal of Graph Theory* 106.1 (2024), pp. 5–22. DOI: 10.1002/jgt.23066.

[PS23] Michał Pilipczuk and Marek Sokołowski. "Graphs of Bounded Twin-Width Are Quasi-Polynomially χ-Bounded". In: *Journal of Combinatorial Theory, Series B* 161 (July 2023), pp. 382–406. DOI: 10.1016/j.jctb.2023.02.006.

[Ram09] F. P. Ramsey. "On a Problem of Formal Logic". In: *Classic Papers in Combinatorics*. Birkhäuser Boston, 2009, pp. 1–24. DOI: 10.1007/978-0-8176-4842-8_1.

[Rap89] David Rappaport. "Computing Simple Circuits from a Set of Line Segments Is NP-Complete". In: *SIAM Journal on Computing* 18.6 (Dec. 1989), pp. 1128–1139. DOI: 10.1137/0218075.

[RIT90] David Rappaport, Hiroshi Imai, and Godfried T. Toussaint. "Computing Simple Circuits from a Set of Line Segments". In: *Discrete & Computational Geometry* 5.3 (June 1990), pp. 289–304. DOI: 10.1007/BF02187791.

[Ree97] B A Reed. "Tree Width and Tangles: A New Connectivity Measure and Some Applications". In: *Surveys in Combinatorics, 1997*. Ed. by R. A. Bailey. London Mathematical Society Lecture Note Series. Cambridge: Cambridge University Press, 1997, pp. 87–162. DOI: 10.1017/CBO9780511662119.006.

[Ree98] B. Reed. "ω, Δ, and χ". In: *Journal of Graph Theory* 27.4 (1998), pp. 177–212. DOI: 10.1002/(SICI)1097-0118(199804)27:4<177::AID-JGT1>3.0.CO;2-K.

[Ree+96] Bruce Reed, Neil Robertson, Paul Seymour, and Robin Thomas. "Packing Directed Circuits". In: *Combinatorica* 16.4 (Dec. 1996), pp. 535–554. DOI: 10.1007/BF01271272.

[Ric46] M. Richardson. "On Weakly Ordered Systems". In: *Bulletin of the American Mathematical Society* 52.2 (Feb. 1946), pp. 113–116. DOI: 10.1090/S0002-9904-1946-08518-3.

[RS95] N. Robertson and P. D. Seymour. "Graph Minors .XIII. The Disjoint Paths Problem". In: *Journal of Combinatorial Theory, Series B* 63.1 (Jan. 1995), pp. 65–110. DOI: 10.1006/jctb.1995.1006.

[Rob+97] Neil Robertson, Daniel Sanders, Paul Seymour, and Robin Thomas. "The Four-Colour Theorem". In: *Journal of Combinatorial Theory, Series B* 70.1 (May 1997), pp. 2–44. DOI: 10.1006/jctb.1997.1750.

[RS86] Neil Robertson and P. D Seymour. "Graph Minors. V. Excluding a Planar Graph". In: *Journal of Combinatorial Theory, Series B* 41.1 (Aug. 1986), pp. 92–114. DOI: 10.1016/0095-8956(86)90030-4.

[RS90] Neil Robertson and P. D Seymour. "Graph Minors. VIII. A Kuratowski Theorem for General Surfaces". In: *Journal of Combinatorial Theory, Series B* 48.2 (Apr. 1990), pp. 255–288. DOI: 10.1016/0095-8956(90)90121-F.

[RS04] Neil Robertson and P. D. Seymour. "Graph Minors. XX. Wagner's Conjecture". In: *Journal of Combinatorial Theory, Series B*. Special Issue Dedicated to Professor W.T. Tutte 92.2 (Nov. 2004), pp. 325–357. DOI: 10.1016/j.jctb.2004.08.001.

[RST93] Neil Robertson, Paul Seymour, and Robin Thomas. "Hadwiger's Conjecture forK 6-Free Graphs". In: *Combinatorica* 13.3 (Sept. 1993), pp. 279–361. DOI: 10.1007/BF01202354.

[RW19] Alexandre Rok and Bartosz Walczak. "Outerstring Graphs Are χ-Bounded". In: *SIAM Journal on Discrete Mathematics* 33.4 (Jan. 2019), pp. 2181–2199. DOI: 10.1137/17M1157374.

[Ros74] M Rosenfeld. "Antidirected Hamiltonian Circuits in Tournaments". In: *Journal of Combinatorial Theory, Series B* 16.3 (June 1974), pp. 234–242. DOI: 10.1016/0095-8956(74)90069-0.

[Ros72] M Rosenfeld. "Antidirected Hamiltonian Paths in Tournaments". In: *Journal of Combinatorial Theory, Series B* 12.1 (Feb. 1972), pp. 93–99. DOI: 10.1016/0095-8956(72)90035-4.

[Roy67] B. Roy. "Nombre chromatique et plus longs chemins d'un graphe". In: *Revue française d'informatique et de recherche opérationnelle* 1.5 (1967), pp. 129–132. DOI: 10.1051/m2an/1967010501291.

[RW08] Ignaz Rutter and Alexander Wolff. "Augmenting the Connectivity of Planar and Geometric Graphs". In: *Electronic Notes in Discrete Mathematics.* The International Conference on Topological and Geometric Graph Theory 31 (Aug. 2008), pp. 53–56. DOI: 10.1016/j.endm.2008.06.009.

[SK07] Amine El Sahili and Mekkia Kouider. "About Paths with Two Blocks". In: *Journal of Graph Theory* 55.3 (2007), pp. 221–226. DOI: 10.1002/jgt.20222.

[Sch13] Jens M. Schmidt. "A Simple Test on 2-Vertex- and 2-Edge-Connectivity". In: *Information Processing Letters* 113.7 (Apr. 2013), pp. 241–244. DOI: 10.1016/j.ipl.2013.01.016.

[SS21a] Thomas Schweser and Michael Stiebitz. "Partitions of Hypergraphs under Variable Degeneracy Constraints". In: *Journal of Graph Theory* 96.1 (2021), pp. 7–33. DOI: 10.1002/jgt.22575.

[SS21b] Thomas Schweser and Michael Stiebitz. "Vertex Partition of Hypergraphs and Maximum Degenerate Subhypergraphs". In: *Electronic Journal of Graph Theory and Applications (EJGTA)* 9.1 (Apr. 2021), pp. 1–9. DOI: 10.5614/ejgta.2021.9.1.1.

[Sco97] A. D. Scott. "Induced Trees in Graphs of Large Chromatic Number". In: *Journal of Graph Theory* 24.4 (1997), pp. 297–311. DOI: 10.1002/(SICI)1097-0118(199704)24:4<297::AID-JGT2>3.0.CO;2-J.

[SS20a] Alex Scott and Paul Seymour. "A Survey of χ-Boundedness". In: *Journal of Graph Theory* 95.3 (2020), pp. 473–504. DOI: 10.1002/jgt.22601.

[SS20b] Alex Scott and Paul Seymour. "Induced Subgraphs of Graphs with Large Chromatic Number. XIII. New Brooms". In: *European Journal of Combinatorics* 84 (Feb. 2020), p. 103024. DOI: 10.1016/j.ejc.2019.103024.

[Sey95] P. D. Seymour. "Packing Directed Circuits Fractionally". In: *Combinatorica* 15.2 (June 1995), pp. 281–288. DOI: 10.1007/BF01200760.

[She02] Jian Shen. "On the Caccetta-Häggkvist Conjecture". In: *Graphs and Combinatorics* 18.3 (Oct. 2002), pp. 645–654. DOI: 10.1007/s003730200048.

[Sku06] San Skulrattanakulchai. "Δ-List Vertex Coloring in Linear Time". In: *Information Processing Letters* 98.3 (May 2006), pp. 101–106. DOI: 10.1016/j.ipl.2005.12.007.

[Spe71] J. Spencer. "Optimal Ranking of Tournaments". In: *Networks* 1.2 (1971), pp. 135–138. DOI: 10.1002/net.3230010204.

[Spe80] J. Spencer. "Optimally Ranking Unrankable Tournaments". In: *Periodica Mathematica Hungarica* 11.2 (June 1980), pp. 131–144. DOI: 10.1007/BF02017965.

[Ste24] Maya Stein. *Oriented Trees and Paths in Digraphs*. May 2024. DOI: 10.48550/arXiv.2310.18719. arXiv: 2310.18719 [math].

[Ste20] Maya Stein. "Tree Containment and Degree Conditions". In: *Discrete Mathematics and Applications*. Ed. by Andrei M. Raigorodskii and Michael Th. Rassias. Vol. 165. Cham: Springer International Publishing, 2020, pp. 459–486. DOI: 10.1007/978-3-030-55857-4_19.

[Ste23] Raphael Steiner. "On Coloring Digraphs with Forbidden Induced Subgraphs". In: *Journal of Graph Theory* 103.2 (2023), pp. 323–339. DOI: 10.1002/jgt.22920.

[SST24] Michael Stiebitz, Thomas Schweser, and Bjarne Toft. *Brooks' Theorem: Graph Coloring and Critical Graphs*. Springer Monographs in Mathematics. Cham: Springer Nature Switzerland, 2024. DOI: 10.1007/978-3-031-50065-7.

[ST18] Michael Stiebitz and Bjarne Toft. "A Brooks Type Theorem for the Maximum Local Edge Connectivity". In: *The Electronic Journal of Combinatorics* (Mar. 2018), P1.50–P1.50. DOI: 10.37236/6043.

[Sul06] Blair Dowling Sullivan. *A Summary of Problems and Results Related to the Caccetta-Haggkvist Conjecture*. May 2006. arXiv: math/0605646.

[Sum81] David P. Sumner. "Subtrees of a Graph and Chromatic Number". In: *The theory and applications of graphs* (1981), pp. 557–576.

[TV85] Robert E. Tarjan and Uzi Vishkin. "An Efficient Parallel Biconnectivity Algorithm". In: *SIAM Journal on Computing* 14.4 (Nov. 1985), pp. 862–874. DOI: 10.1137/0214061.

[Tho10] Stéphan Thomassé. "A 4k2 Kernel for Feedback Vertex Set". In: *ACM Trans. Algorithms* 6.2 (Apr. 2010), 32:1–32:8. DOI: 10.1145/1721837.1721848.

[Tho83] Carsten Thomassen. "Disjoint Cycles in Digraphs". In: *Combinatorica* 3.3 (Sept. 1983), pp. 393–396. DOI: 10.1007/BF02579195.

[TZ23] Vera Traub and Rico Zenklusen. "A (1.5+ ε)-Approximation Algorithm for Weighted Connectivity Augmentation". In: *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*. 2023, pp. 1820–1833. DOI: 10.1145/3564246.3585122.

[Tur41] Pal Turan. "On an Extremal Problem in Graph Theory". In: *Mat. Fiz. Lapok* 48 (1941), pp. 436–452.

[UV22] Mario Ullrich and Jan Vybíral. "Deterministic Constructions of High-Dimensional Sets with Small Dispersion". In: *Algorithmica* 84.7 (July 2022), pp. 1897–1915. DOI: 10.1007/s00453-022-00943-x.

[Vég11] László A. Végh. "Augmenting Undirected Node-Connectivity by One". In: *SIAM Journal on Discrete Mathematics* 25.2 (Jan. 2011), pp. 695–718. DOI: 10.1137/100787507.

[VB08] László A. Végh and András A. Benczúr. "Primal-Dual Approach for Directed Vertex Connectivity Augmentation and Generalizations". In: *ACM Trans. Algorithms* 4.2 (May 2008), 20:1–20:21. DOI: 10.1145/1361192.1361197.

[Vit62] L. M. Vitaver. "Determination of Minimal Coloring of Vertices of a Graph by Means of Boolean Powers of the Incidence Matrix". In: *Dokl. Akad. Nauk SSSR* 147.4 (1962), pp. 758–759.

[vMR44] John von Neumann, Oskar Morgenstern, and Ariel Rubinstein. *Theory of Games and Economic Behavior (60th Anniversary Commemorative Edition)*. Princeton University Press, 1944. JSTOR: j.ctt1r2gkx.

[Wal72] David W. Walkup. "The Number of Plane Trees". In: *Mathematika. A Journal of Pure and Applied Mathematics* 19.2 (1972), pp. 200–204. DOI: 10.1112/s0025579300005659. eprint: https://londmathsoc.onlinelibrary.wiley.com/doi/pdf/10.1112/S0025579300005659.

[WN90] Toshimasa Watanabe and Akira Nakamura. "A Smallest Augmentation to 3-Connect a Graph". In: *Discrete Applied Mathematics* 28.2 (Aug. 1990), pp. 183–186. DOI: 10.1016/0166-218X(90)90116-T.

[WN87] Toshimasa Watanabe and Akira Nakamura. "Edge-Connectivity Augmentation Problems". In: *Journal of Computer and System Sciences* 35.1 (Aug. 1987), pp. 96–144. DOI: 10.1016/0022-0000(87)90038-9.

[Wor83] Nicholas C. Wormald. "Subtrees of Large Tournaments". In: *Combinatorial Mathematics X*. Ed. by Louis Reynolds Antoine Casse. Berlin, Heidelberg: Springer, 1983, pp. 417–419. DOI: 10.1007/BFb0071535.

[You73] DH Younger. "Graphs with Interlinked Directed Circuits". In: *Proceedings of the Midwest Symposium on Circuit Theory*. Vol. 2. 1973, pp. XVI–2.

[Yus25] Raphael Yuster. "On Tournament Inversion". In: *Journal of Graph Theory* 110.1 (2025), pp. 82–91. DOI: 10.1002/jgt.23251.

[Zyk49] A. A. Zykov. "On some properties of linear complexes". In: *Matematicheskii Sbornik. Novaya Seriya* 24.2 (1949), pp. 163–188.

Index