Graph sharing games: strategies and algorithms

Bartosz Walczak

Ph.D. Thesis
Theoretical Computer Science Department
Faculty of Mathematics and Computer Science
Jagiellonian University
advised by Paweł M. Idziak

Kraków, 2012

Abstract

Graph sharing games are two-player combinatorial games played on connected graphs with non-negative weight assigned to the vertices. Two players take the vertices alternately one by one collecting their weights. They must obey some rules restricting the availability of vertices. The game ends when all vertices have been taken. The goal of both players is to collect as much weight as possible.

We consider two kinds of restrictions on the availability of vertices, leading to two variants of graph sharing games: game T and game R. The rule of game T is that the set of taken vertices after each move forms a connected subgraph of the original graph. In game R, every move must lead to a position at which the set of remaining vertices forms a connected subgraph of the original graph.

We present bounds on the gain that the first player can guarantee himself in game T and game R on graphs with restricted structure and parity of the number of vertices. One of our main results says that for every n there is a positive constant bounding from below the first player's guaranteed outcome (computed as a fraction of the total weight) in game T on graphs with an odd number of vertices and with forbidden subdivision of the n-vertex clique. For $n = 3$, that is, for odd trees, this constant is $1/4$. We also prove that the first player can secure at least $1/4$ of the total weight in game R played on even trees. Furthermore, we provide constructions of graphs proving (for both variants of the game) that positive lower bounds for the first player's guaranteed relative outcome cannot exist unless both the structure and the parity of the number of vertices are restricted.

As a byproduct of the result about game T on graphs with forbidden subdivision, we obtain a new characterization of such graphs, which may deserve independent interest. Our second main achievement is a polynomial-time algorithm computing the optimal game value for game T played on trees.

Acknowledgments

Graph sharing games originated as a generalization of Peter Winkler's problem of sharing a pizza, after it had been solved by Kolja Knauer, Piotr Micek, and Torsten Ueckerdt. Since then, I have been working on graph sharing games together with Piotr Micek and Adam Gągol. I thank them for inspiration, motivation, and hours of fruitful discussions, which led to many of the results contained in this thesis. In particular, the results of Chapters 5 and 7 have been obtained jointly with Piotr Micek, while the results of Chapter 8 are based on earlier joint work with Adam Gągol and Piotr Micek on classes of graphs with forbidden minors, contained in Adam's master thesis [6]. I am especially grateful to Piotr for introducing me to the topic just when it arose, and to Adam for his brilliant ideas that enabled us to tackle the problems of sharing graphs with forbidden minors or subdivisions. I also thank Jan Hązła, Tomasz Idziaszek, Tomasz Krawczyk, and Torsten Ueckerdt for helpful discussions and comments. Finally, I thank my advisor Paweł M. Idziak for scientific guidance during my entire Ph.D. studies and for many hours spent on correcting draft versions of this thesis.

The author has been granted scholarship in the project *Doctus – Małopolski fundusz stypendialny dla doktorantów* co-funded by the European Union through the European Social Fund.

Contents

Chapter 1

Introduction

Combinatorial game theory has broad applications in computer science, mostly in on-line algorithms, distributed systems, and artificial intelligence. The classical part of the theory deals with games having two outcome classes: one player wins and the other loses. Such games are well studied and characterized. Many natural combinatorial games, however, do not fit into this classical setting. Good examples are games in which the players compete for valuable resources and the result is the sum of the values collected by each of the players. Graph sharing games constitute a model for games of this kind equipped with graph structure underlying the resource supply. They have been recently popularized by Winkler [18] and Rosenfeld [15]. As the research on these games developed, it has revealed their intriguing properties and has lead to interesting graph-theoretic and algorithmic considerations going far beyond original expectations. This thesis presents several results on combinatorial and algorithmic aspects of graph sharing games, obtained by the author in cooperation with Adam Gągol and Piotr Micek.

Graph sharing games are played on a finite connected graph with non-negative weights on the vertices. There are two players: Alice and Bob. Starting with Alice, they take the vertices alternately one by one and collect their weights. The vertices taken are removed from the graph. The choice of a vertex to be played in each move is restricted depending on the variant of the game:

- In game T the rule is that after each move the vertices taken so far form a connected subgraph of the original graph;
- In game R the rule is that after each move the remaining vertices form a connected subgraph.

The game ends when all the vertices have been taken. Both players' goal is to maximize the total weight they have gathered at the end.

The two variants of the game can be totally different in general. However, they coincide when the graph is a cycle. This case has been studied as the so-called pizza game: vertices are seen as slices of a pizza. In 1996 Brown asked whether Alice has a strategy to collect at least $\frac{1}{2}$ of the weight of any pizza. This can be easily confirmed for pizzas with an even number of slices: color alternately the slices with two colors and secure the heavier color. At first glance the case of pizzas with an odd number of slices looks better for Alice as she gets one slice more than Bob. Curiously things can get worse for her: there are examples where she can get only $\frac{4}{9}$ (see Figure 1.1). Winkler [18] conjectured that Alice can secure at least $\frac{4}{9}$ of any pizza. This has been proved by two independent groups of researchers.

Theorem 1.1 (Cibulka, Kynčl, Mészáros, Stolař, Valtr [2]; Knauer, Micek, Ueckerdt [10])**.** *Alice can secure at least $\frac{4}{9}$ of the total weight of any cycle.*

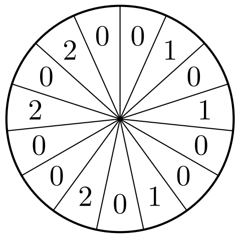

Figure 1.1. Alice can get at most $\frac{4}{9}$ of the pizza playing against clever Bob. Numbers stand for slice weights.

Rosenfeld [15] considered game R on trees, vaguely asking for good strategies for Alice. Both variants of the game for general graphs have been introduced by Cibulka, Kynčl, Mészáros, Stolař, and Valtr [3] and independently by Micek and the author [13, 14].

As the the case of cycles suggests, the right parameter to measure for graph sharing games is Alice's relative outcome rather than her absolute outcome. That is, we are interested in the weight Alice secures divided by the total weight of the graph. A natural question arises for both variants of the game: Is there a universal constant $c > 0$ bounding from below Alice's relative outcome on any graph? The answer is that no such constant exists even for graphs from very narrow classes. Examples presented in the next chapter show that such a lower bound cannot exist for either variant even if the game is played on trees instead of arbitrary graphs. However, if the parity of the number of vertices is restricted (odd for game T, even for game R), the situation looks very different. The following results are to be presented in this thesis.

Theorem 1.2 ([14])**.** *Alice can secure at least $\frac{1}{4}$ of the total weight of any tree with an odd number of vertices in game* T*.*

Theorem 1.3. *For every n there is a constant $c_n > 0$ such that Alice can secure in game* T *at least c_n of the total weight of any graph with an odd number of vertices containing no subdivision of K_n.*

Theorem 1.4 ([13])**.** *Alice can secure at least $\frac{1}{4}$ of the total weight of any tree with an even number of vertices in game* R*.*

Theorem 1.3 generalizes the following result that has been the subject of Gągol's master thesis.

Theorem 1.5 ([6])**.** *For every n there is a constant $c_n > 0$ such that Alice can secure in game* T *at least c_n of the total weight of any graph with an odd number of vertices, weights from the set $\{0, 1\}$, and no K_n minor.*

On the other hand, Example 2.2 shows that Alice's relative outcome in game T on a graph with an odd number of vertices can be arbitrarily small. Subdivision of a large clique is a structure naturally arising in the construction presented in Example 2.2. Moreover, this construction can be improved to show that Alice's relative outcome in game T can be arbitrarily small on graphs with an odd number of vertices restricted to a class of graphs with bounded expansion. Such classes are the next classes of graphs established in the literature broader than excluding a subdivision of a fixed clique. Thus in a sense Theorem 1.3 is best possible and cannot be generalized further.

To prove Theorem 1.3 we show a structural characterization of graphs with forbidden subdivision, which may be worth independent interest. Essentially it says that every weighted connected graph G with no subdivision of K_n contains one of the following heavy structures for some $c_n > 0$ depending only on n:

- a connected set separating the graph into components all of which except the heaviest one have total weight at least $c_n \cdot w(G)$;
- a set of vertices of total weight at least $c_n \cdot w(G)$ connected by paths like in a cycle.

This result is presented in detail in Chapter 8.

Theorem 1.4 has been superseded by the following result confirming our conjecture from [13].

Theorem 1.6 (Seacrest, Seacrest [16])**.** *Alice can secure at least $\frac{1}{2}$ of the total weight of any tree with an even number of vertices in game* R*.*

It is clear that the constant $\frac{1}{2}$ above is best possible. No class of graphs with an even number of vertices significantly broader than the class of trees in known for which Alice's relative outcome in game R is still bounded away from zero. On the other hand, Example 2.3 shows that Alice's relative outcome in game R on a graph with an even number of vertices can be arbitrarily small. We have no idea what the border line separating these two cases can look like.

The other type of problems considered in this thesis concern efficient algorithms computing optimal strategies of both players in graph sharing games. This topic has been brought up by Cibulka et al. [3]. They have proved that deciding which player has a strategy to gather more than $\frac{1}{2}$ of the weight of a given graph in game R is PSPACE-complete. Whether the same holds for game T is left open. They have also asked about the computational complexity of determining the winner in game T and game R played on trees. We answer this question for game T.

Theorem 1.7. *There is a polynomial-time algorithm computing optimal strategies of both players in game* T *on trees.*

The thesis is organized as follows. In the next chapter we present constructions of graphs bounding Alice's guaranteed relative outcome from above in game T and game R on several classes of graphs. In Chapter 3 we introduce the terminology and notation, as well as some known graph-theoretical results that are necessary for our proofs. In Chapter 4 we define our games formally, first in an abstract setting and then in the graph setting. Chapter 5 contains the proof of Theorem 1.2. In Chapter 6 we present the proof of Theorem 1.7. Chapter 7 contains the proof of Theorem 1.4. Chapter 8 contains the proof of the structural characterization of graphs with forbidden subdivision, necessary for the proof of Theorem 1.3. In Chapter 9 we prove Theorem 1.3. We conclude in the last chapter with some open problems.

Chapter 2

Examples

Before we define the games formally, we present several examples demonstrating that Alice may be unable to gather a substantial fraction of the total weight of the graph in game T or game R.

2.1. Game T

At the start of game T played on a graph G all vertices of G are available for Alice. She can pick the heaviest vertex thus securing at least $1/|V(G)|$ of the total weight of G with her first move. In general she cannot be sure to get much more. The following example was presented to us by Kierstead [8].

Example 2.1. Consider the weighted graph G_n with $2n$ vertices $a_1, \ldots, a_n$, $b_1, \ldots, b_n$ and edge set

$$E(G_n) = \{a_1b_1, \ldots, a_nb_n\} \cup \{b_ib_j : i \neq j\}.$$

The weight is 1 on every a_i and 0 on every b_i. Thus the total weight is n. See Figure 2.1 for an illustration of G_5.

Alice has no strategy to gather more than 1 from G_n in game T. Indeed, if she starts with a_i or b_i then Bob responds by taking the other vertex of a_i, b_i. Then, in all subsequent moves Alice is forced to take some vertex of the clique, say b_j, and Bob responds by playing a_j. □

In the example above the clique on $b_1, \ldots, b_n$ can be replaced by any connected graph (a path, a star, etc.), and the argument continues to work. This shows that even for very simple classes of graphs (caterpillars, subdivided stars) Alice's guaranteed relative outcome tends to zero as the size of the graph goes to infinity. However, all graphs constructed this way have an even number of vertices. Things get more complicated if we ask for a sequence of graphs with an *odd* number of vertices and arbitrarily small Alice's guaranteed relative outcome. The following construction has been also found by Valtr and his students [17].

Example 2.2. Consider the weighted graph H_n with $2n + 2^n - 1$ vertices

$$V(H_n) = \big\{a_1, \ldots, a_n, b_1, \ldots, b_n\big\} \cup \big\{c_X : X \subseteq \{b_1, \ldots, b_n\} \text{ and } X \neq \emptyset\big\}$$

and edge set

$$E(H_n) = \{a_1b_1, \ldots, a_nb_n\} \cup \{b_ic_X : i \in X\}.$$

The weight is 1 on every a_i, 0 on every b_i, and 0 on every c_X. Thus the total weight is n. See Figure 2.2 for an illustration.

We show that Alice can secure at most 1 in game T on H_n. For the proof suppose that Alice starts with a_{i_1} or b_{i_1}. Then Bob responds by taking the other of a_{i_1}, b_{i_1}. If $n - 1 > 0$ then define

$$V_1 = \{a_i, b_i : i \neq i_1\} \cup \{c_X : b_{i_1} \notin X\}$$

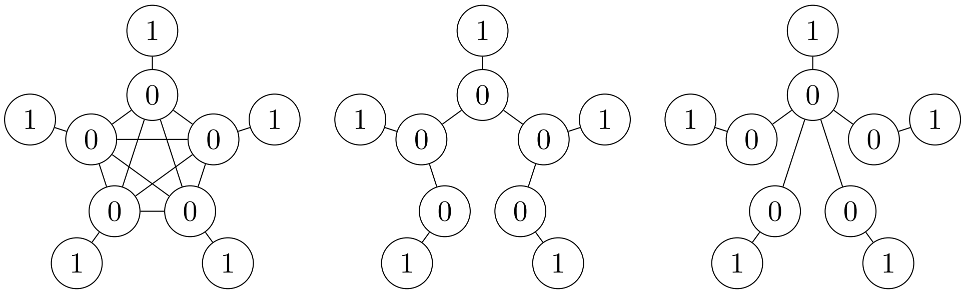

Figure 2.1. G_5 and related examples

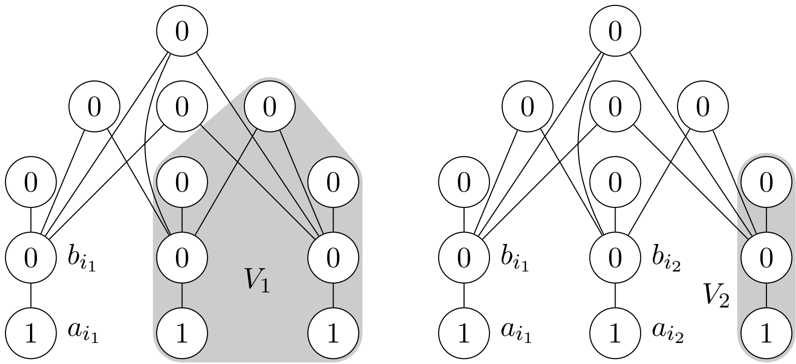

Figure 2.2. H_3: Alice is forced to enter sets V_1 and V_2.

Note that the subgraph of H_n induced by V_1 is isomorphic to H_{n-1}. In particular, $|V_1|$ is odd and $|V - V_1|$ is even. Since b_{i_1} is taken, all vertices in $V - V_1$ are available. Therefore, as long as Alice plays in $V - V_1$, Bob can respond also in $V - V_1$. Alice is eventually forced to enter V_1, which is possible only by taking some b_{i_2}, and Bob immediately follows with a_{i_2}. If $n - 2 > 0$ then define

$$V_2 = \{a_i, b_i : i \neq i_1, i_2\} \cup \{c_X : b_{i_1}, b_{i_2} \notin X\}$$

and continue with the same argument. And so on. This way Bob wins every vertex a_i other than a_{i_1}. If Alice starts with some c_X then Bob takes any available b_{i_1} and the same argument shows that Bob can take all vertices a_i other than a_{i_1}. □

Note that the paths of length 2 connecting b_i and b_j through $c_{\{b_i,b_j\}}$ form a subdivision of K_n in H_n. When we replace every edge $b_i c_X$ in H_n by a path of length $2n+1$ connecting b_i and c_X, we obtain a sequence of graphs with an odd number of vertices, bounded expansion (see Section 3.2), and Alice's guaranteed outcome tending to zero.

Cibulka et al. [3] constructed a sequence of k-connected graphs with Alice's guaranteed relative outcome in game T tending to zero for any given k. They also observed that Examples 2.1 and 2.2 lead to such sequences consisting of k-connected graphs of either parity: just replace each original 0-vertex by an odd clique of 0-vertices and each original edge by a complete bipartite graph.

The best upper bound on Alice's guaranteed relative outcome we know for odd trees is $\frac{2}{5}$, realized by the tree in Figure 2.3.

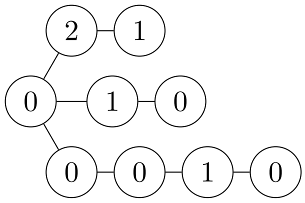

Figure 2.3. Alice gets at most 2 out of 5 on this tree in game T.

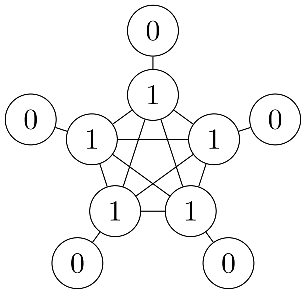

Figure 2.4. G'_5: Alice cannot secure more than 1 in game R.

2.2. Game R

For game R there are graphs on which Alice cannot guarantee herself any positive outcome. The simplest example of such a graph is a 3-vertex path with all the weight at the middle vertex. We can also construct graphs with an even number of vertices and arbitrarily small Alice's guaranteed relative outcome, as follows.

Example 2.3. Consider the weighted graph G'_n with $2n$ vertices $a_1, \ldots, a_n$, $b_1, \ldots, b_n$ and edge set

$$E(G_n) = \{a_1 b_1, \ldots, a_n b_n\} \cup \{b_i b_j : i \neq j\}.$$

The weight is 0 on every a_i and 1 on every b_i. Thus the total weight is again n. See Figure 2.4 for an illustration of G'_5.

Alice has no strategy to gather more than 1 from G'_n in game R. She cannot take any b_i when the corresponding a_i is still available unless a_i and b_i are the only remaining vertices. Whenever she takes a vertex a_i, Bob responds by taking b_i. It follows that Alice can take a vertex b_i only with her last move. $\square$

Note that unlike in Example 2.1, the clique in the example above cannot be replaced by any proper subgraph.

Cibulka et al. [3] constructed a sequence of k-connected graphs with an even number of vertices and with Alice's guaranteed relative outcome in game R tending to zero for any given k. They also gave a probabilistic construction of graphs with an even number of vertices that contain no K_3 and can be arbitrarily bad for Alice. We do not know any construction of even graphs with bounded chromatic number and arbitrarily small guaranteed relative outcome of Alice.

Chapter 3

Background

In this chapter we introduce the terminology, notation, and results from graph theory that we use later in the thesis. The terminology and notation related to games is explained in the next chapter.

3.1. Basic terminology and notation

We denote by $\mathbb{N}$ the set of non-negative integers and by $\mathbb{N}^+$ the set of positive integers. The size of a finite set X is denoted by $|X|$. If X is a finite set and $w : X \to \mathbb{R}$ then we define $w(X)$ by

$$w(X) = \sum_{x \in X} w(x).$$

In our terminology a *graph* is an unoriented graph unless explicitly stated otherwise. A graph G consists of a finite set of *vertices*, denoted by $V(G)$, and a set of unoriented *edges* connecting pairs of vertices, denoted by $E(G)$. An edge connecting vertices $u, v \in V(G)$ is denoted by uv and is the same as the edge vu. The graphs we consider have no loops nor multiple edges. If $uv \in E(G)$ then we call u and v *adjacent*. For a vertex $v \in V(G)$ we define

- $N_G(v)$, the *neighborhood* of v, to be the set of vertices adjacent to v;
- $N_G[v]$, the *closed neighborhood* of v, to be $N_G(v) \cup \{v\}$;
- $d_G(v)$, the *degree* of v, to be $|N_G(v)|$.

A graph H is a *subgraph* of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. If H is a subgraph in G then we say simply that H *is* in G or that G *contains* H. For a set $S \subseteq V(G)$, we call a vertex $v \in V(G) - S$ *adjacent* to S if it is adjacent to some vertex in S, and we define

- $N_G(S)$, the *neighborhood* of S, to be the set of vertices from $V(G) - S$ adjacent to S;
- $N_G[S]$, the *closed neighborhood* of S, to be $N_G(S) \cup S$;
- $G[S]$, the subgraph *induced* on S, to be the graph with vertex set S and edge set $\{uv \in E(G) : u, v \in S\}$;
- $G \cap S = G[V(G) \cap S]$ and $G - S = G[V(G) - S]$.

For a graph G we define

$$\nabla G = \max_{S \subseteq V(G),\, S \neq \emptyset} \frac{|E(G[S])|}{|S|}.$$

For two graphs G_1 and G_2 we define

- $G_1 \cup G_2$ to be the graph with vertex set $V(G_1) \cup V(G_2)$ and edge set $E(G_1) \cup E(G_2)$;
- $G_1 \cap G_2$ to be the graph with vertex set $V(G_1) \cap V(G_2)$ and edge set $E(G_1) \cap E(G_2)$.

Two graphs G_1 and G_2 are *isomorphic* if there is a bijection $\phi : V(G_1) \to V(G_2)$ such that $uv \in E(G_1)$ if and only if $\phi(u)\phi(v) \in E(G_2)$ for any $u, v \in V(G_1)$.

A *path* of length k is a graph consisting of $k+1$ vertices $v_0, \ldots, v_k$ and k edges $v_0v_1, v_1v_2, \ldots, v_{k-1}v_k$. It is denoted by $v_0 \ldots v_k$. Such a path is called *connecting* the vertices v_0 and v_k. The vertices v_0 and v_k are the *ends* of the path, while $v_1, \ldots, v_{k-1}$ are its *internal vertices.* A graph G is *connected* if for any $u, v \in V(G)$ there is a path in G connecting u and v. A subset S of $V(G)$ is called *connected* in G if $G[S]$ is connected. When it is clear from the context in which graph the set is connected, we do not specify this explicitly and just call a subset connected. A *component* of G is a maximal connected subgraph of G. The family of components of G is denoted by $\operatorname{Comp} G$. The *distance* between $u, v \in V(G)$, denoted by $\operatorname{dist}_G(u, v)$, is the minimum length of a path connecting u and v in G. If there is no such path then $\operatorname{dist}_G(u, v) = \infty$. The *radius* of a set $S \subseteq V(G)$, denoted by $r_G(S)$, is defined by

$$r_G(S) = \min_{u \in S} \max_{v \in S} \operatorname{dist}_G(u, v).$$

If S is not connected then $r_G(S) = \infty$.

We call two disjoint sets $S_1, S_2 \subseteq V(G)$ *adjacent* if some vertex in S_1 is adjacent to some vertex in S_2. For a partition $\mathcal{S}$ of $V(G)$ into connected non-empty subsets, we define $G/\mathcal{S}$ to be the graph with vertex set $\mathcal{S}$ and edge set defined by the adjacency relation on subsets of $V(G)$.

For a graph G and a set $S \subseteq V(G)$ we denote by $G\{S\}$ the graph with vertex set S and edge set defined as follows: $uv \in E(G\{S\})$ if there is a path P in G connecting u and v internally disjoint from S, that is, such that $V(P) \cap S = \{u, v\}$. If G is connected then $G\{S\}$ is connected as well for any $S \subseteq V(G)$.

We define a *cycle* as follows. A cycle of length $k \geqslant 3$ is a graph consisting of k vertices $v_0, \ldots, v_{k-1}$ and k edges $v_0v_1, \ldots, v_{k-2}v_{k-1}, v_{k-1}v_0$. A graph consisting of a single vertex is a cycle of length 1, while a graph with two vertices joined by an edge is a cycle of length 2. With this unusual definition for lengths 1 and 2 we do not need to consider degenerate cases in Chapters 8 and 9.

An n-vertex *clique* is a graph with n vertices and $\binom{n}{2}$ edges connecting all pairs of vertices. We denote by K_n a clique on any set of n vertices.

An *oriented graph* is a graph in which every edge is assigned an orientation from one to the other of its ends. We write uv for an edge connecting u and v to denote that it is oriented from u to v. Every subgraph of an oriented graph inherits orientation of all edges and thus is also an oriented graph. An oriented path is a path $v_0 \ldots v_k$ such that $v_0v_1, v_1v_2, \ldots, v_{k-1}v_k$ are oriented edges. Such a path is called a *v_0, v_k-path.* The vertices v_0 and v_k are respectively the *begin* and the *end* of a v_0, v_k-path. An oriented cycle is a cycle of length $k \geqslant 3$ consisting of oriented edges $v_0v_1, \ldots, v_{k-2}v_{k-1}, v_{k-1}v_0$.

A *rooted graph* is a graph, in which some vertices (at least one) are selected as *roots.* The set of roots in a rooted graph G is denoted by $R(G)$. If G is a graph and $S \subseteq V(G)$ then we often consider $G[S]$ as a rooted graph with root set $N_G(V(G) - S)$. Consequently, if G is a rooted graph and $S \subseteq V(G)$ then we consider $G[S]$ as a rooted graph with root set $(R(G) \cap S) \cup N_G(V(G) - S)$.

A graph G (ordinary, oriented, or rooted) can be equipped with an additional function $w : V(G) \to [0, \infty)$ that assigns a *weight* to each vertex of G. We call such a graph *weighted*. If $S \subseteq V(G)$ then $w(S)$ denotes the sum of weights of the vertices in S. We denote by $w(G)$ the sum of weights of all vertices in G. If H is a subgraph of G then H inherits the weight function w restricted to $V(H)$ unless explicitly stated otherwise. Similarly, if $S \subseteq V(G)$ then $G\{S\}$ inherits the weight function w restricted to S.

A *tree* is a connected graph containing no cycle of length greater than 2. For every tree T we have $|E(T)| = |V(T)| - 1$. Any two vertices u and v in any tree T are connected by a unique path denoted by T_{uv}. A *leaf* in an unrooted tree is a vertex of degree 1. A *forest* is a graph with every component being a tree. A rooted tree is implicitly assumed to have exactly one root. Consequently, a rooted forest contains exactly one root in each component. For a rooted tree T with root r and a vertex $u \in V(T)$ we denote by $T(u)$ the rooted subtree of T with vertex set $\{v \in V(T) : u \in V(T_{rv})\}$ and root u. The *children* of a vertex u of a rooted tree T are the neighbors of u in $T(u)$. A *leaf* in a rooted tree is a vertex with no children. A rooted tree T with root r can be naturally oriented so that every vertex v is reachable from r by the oriented path T_{rv}. Then, for a vertex $u \in V(T)$, the chilren of u are the end-vertices of the edges going out of u, and the subtree $T(u)$ consists of those vertices of T that are reachable from u by oriented paths. For a rooted forest F and a vertex $u \in V(F)$ we denote by $F(u)$ the rooted subtree $T(u)$, where T is the component of F containing u. Children and leaves in a rooted forest are defined just like in a rooted tree.

A vertex v of a weighted tree T is called a *weighted center* of T if every component of $T - \{v\}$ has weight at most $\frac{1}{2}w(T)$.

Proposition 3.1. *Every weighted tree has a weighted center.*

Proof. Let T be a weighted tree. Pick any vertex $v_0 \in V(T)$. Then, for each i, either v_i is a weighted center or exactly one component C of $T - \{v_i\}$ has weight greater than $\frac{1}{2}w(T)$. In the latter case choose the only neighbor of v_i in C to be v_{i+1}. This way a simple path $v_0 v_1 \ldots$ is constructed. It cannot be infinite, so a weighted center of T is finally found. □

A tree T is a *spanning tree* of a graph G if T is a subgraph of G and $V(T) = V(G)$. An edge $uv \in E(G) - E(T)$ is a *crossing edge* with respect to a spanning tree T of G and a vertex $r \in V(G)$ if $u \notin V(T_{rv})$ and $v \notin V(T_{ru})$. A *depth-first search tree* in G with respect to a vertex $r \in V(G)$ is a spanning tree T such that G has no crossing edges with respect to T and r.

Proposition 3.2 (see e.g. [4] p. 16)**.** *Every connected graph G has a depth-first search tree with respect to any $r \in V(G)$.*

A depth-first search tree can be found in linear-time by the algorithm called depth-first search (hence the name).

3.2. Subdivisions and minors

A graph F is a *subdivision* of a graph H if it is obtained from H as a result of replacing every edge $uv \in E(H)$ by a path F_{uv} internally disjoint from $V(H)$ and any other path $F_{u'v'}$. The relation of being a subdivision is transitive: if F_1 is a subdivision of F_2 and F_2 is a subdivision of H then F_1 is also a subdivision of H.

Theorem 3.3 (Mader [12])**.** *For every graph H there is p_H such that every graph G containing no subdivision of H satisfies $\nabla G \leqslant p_H$.*

Komlós and Szemerédi [11] and independently Bollobás and Thomason [1] have proved that ∇G for graphs G containing no subdivision of K_n is bounded by $O(n^2)$. This is best possible: there are graphs G_n containing no subdivision of K_n such that $\nabla G_n = \Theta(n^2)$ [7].

A *minor* of G is a subgraph of $G/\mathcal{S}$ for a partition $\mathcal{S}$ of $V(G)$ into connected non-empty subsets. Equivalently, a graph H is a minor of G if there is a family $\{S(v)\}_{v\in V(H)}$ of pairwise disjoint connected non-empty subsets of $V(G)$ such that $S(u)$ and $S(v)$ are adjacent whenever $uv \in E(H)$. Every subgraph of G is a trivial minor of G. The relation of being a minor is transitive: if H_1 is a minor of H_2 and H_2 is a minor of G then H_1 is also a minor of G. If G contains a subdivision of H then H is a minor of G.

An *r-shallow minor* of G is a subgraph of $G/\mathcal{S}$ for a partition $\mathcal{S}$ of $V(G)$ into connected non-empty subsets of radius at most r. A 0-shallow minor of G is just a subgraph of G. Every minor of G is an r-shallow minor for some r. We define

$$\nabla_r G = \max_{H\in\mathcal{M}_r(G)} \nabla H = \max_{H\in\mathcal{M}_r(G)} \frac{|E(H)|}{|H|},$$

where $\mathcal{M}_r(G)$ denotes the family of all r-shallow minors of G.

Theorem 3.4 (Dvořák [5]). *For any $r, d \in \mathbb{N}$ there is $p_{r,d}$ such that every graph G with $\nabla_r G \geqslant p_{r,d}$ contains a subdivision of a graph H with minimum degree d.*

Corollary 3.5. *For any $r \in \mathbb{N}$ and $n \in \mathbb{N}^+$ there is $p_{r,n}$ such that every graph G with $\nabla_r G \geqslant p_{r,n}$ contains a subdivision of K_n.*

Proof. Fix $r \in \mathbb{N}$ and $n \in \mathbb{N}^+$. Let p' be a constant claimed by Theorem 3.3 for K_n. Choose $d \in \mathbb{N}$ so that $d/2 > p'$. Let $p_{r,n}$ be a constant claimed by Theorem 3.4 for r and d. Thus every graph G with $\nabla_r G \geqslant p_{r,n}$ contains a subdivision of a graph H with minimum degree d. For such H we have $\nabla H \geqslant d/2 > p'$, which by Theorem 3.3 implies that H (and thus G) contains a subdivision of K_n. □

Corollary 3.6. *For any $r \in \mathbb{N}$ and $n \in \mathbb{N}^+$ there is $N \in \mathbb{N}^+$ such that if G contains no subdivision of K_n then every r-shallow minor of G contains no subdivision of K_N.*

Proof. Fix $r \in \mathbb{N}$ and $n \in \mathbb{N}^+$. Let $p_{r,n}$ be a constant claimed by Corollary 3.5. Choose $N \in \mathbb{N}^+$ so that

$$\frac{N-1}{2} \geqslant p_{r,n}.$$

We show that if an r-shallow minor H of G contains a subdivision of K_N then G contains a subdivision of K_n.

Let $\{S(v)\}_{v\in H}$ be a family of subsets of $V(G)$ witnessing H as an r-shallow minor of G, that is, a family of pairwise disjoint connected non-empty subsets of $V(G)$ of radius at most r such that $S(u)$ and $S(v)$ are adjacent whenever $uv \in E(H)$. Let F be a subdivision of K_N in H. This means that there are N vertices $v_1, \ldots, v_N$ and $\binom{N}{2}$ paths F_{ij} (for $i, j \in \{1, \ldots, N\}$, $i \neq j$) in F such that each F_{ij} connects v_i with v_j and is internally disjoint from $\{v_1, \ldots, v_N\}$ and any other path $F_{i'j'}$. Define

$$S_i = S(v_i), \qquad S_{ij} = \bigcup_{v\in V(F_{ij})} S(v) \quad \text{for } i \neq j, \qquad S = \bigcup_{i=1}^{N} S_i.$$

By the property of the family $\{S(v)\}_{v\in H}$ the sets S_{ij} are connected. Therefore, for $i \neq j$ there is a path in $G[S_{ij}]$ connecting a vertex from S_i with a vertex from S_j. Let P_{ij} be a shortest such path. It follows that P_{ij} is internally disjoint from S, and thus every internal vertex of P_{ij} belongs to $S(v)$ for an internal vertex v of F_{ij}. Therefore, every path P_{ij} is internally disjoint from any other $P_{i'j'}$.

Let G' be the union of the graphs $G[S_i]$ and the paths P_{ij}. Let $G^\star$ be the graph with vertex set S obtained from G' by replacing every path P_{ij} by a single edge connecting its ends. The sets S_i and S_j are adjacent in $G^\star$ for $i \neq j$. Therefore, K_N is an r-shallow minor of $G^\star$. It follows that

$$\nabla_r G^\star \geqslant \nabla K_N = \frac{N-1}{2} \geqslant p_{r,n}.$$

Thus by Corollary 3.5 the graph $G^\star$ contains a subdivision $F^\star$ of K_n. Replace every edge of $F^\star$ that connects S_i and S_j with $i \neq j$ by the path P_{ij}. This results in a subdivision of $F^\star$ (and hence of K_n) in G' (and hence in G). We have thus proved that G contains a subdivision of K_n. □

A class of graphs $\mathcal{G}$ has *bounded expansion* if there is a function $f : \mathbb{N} \to \mathbb{N}$ such that $\nabla_r G \leqslant f(r)$ for every $r \in \mathbb{N}$ and every $G \in \mathcal{G}$. By Corollary 3.5 the class of graphs containing no subdivision of K_n has bounded expansion.

3.3. Vertex orderings and colorings

We denote by $\Pi(S)$ the family of all linear orderings of a set S. For $\pi \in \Pi(S)$ and $x \in S$ we define

$$\pi^-(x) = \big\{y \in S : y <_\pi x\big\}, \qquad \pi^+(x) = \big\{y \in S : y >_\pi x\big\}.$$

Following [9], for $k \in \mathbb{N}^+$ we call a vertex u *weakly k-accessible* from a vertex v in an ordering $\pi \in \Pi(V(G))$ if $u <_\pi v$ and there is a path from v to u of length at most k disjoint from $\pi^-(u)$, and we define

$$\operatorname{wcol}_k G = \min_{\pi\in\Pi(V(G))} \max_{v\in V(G)} |Q_\pi(v)| + 1,$$

where $Q_\pi(v)$ denotes the set of vertices of G weakly k-accessible from v in π.

Proposition 3.7. *For every graph G and every ordering $\pi \in \Pi(V(G))$ realizing $\operatorname{wcol}_k G$, the vertices of G can be colored using at most $\operatorname{wcol}_k G$ colors so that no vertex is weakly k-accessible in π from another vertex of the same color.*

Proof. Let C be a set of colors of size $\operatorname{wcol}_k G$. Process the vertices according to the order π, assigning each vertex v a color from C different from the colors already assigned to the vertices in $Q_\pi(v)$. At least one such color is available as $|Q_\pi(v)| < \operatorname{wcol}_k G = |C|$. □

Theorem 3.8 (Kierstead, Yang [9]). *For any graph H and $k \in \mathbb{N}^+$ there is $d_{H,k}$ such that every graph G containing no subdivision of H satisfies $\operatorname{wcol}_k G \leqslant d_{H,k}$.*

Chapter 4

Sharing games

This thesis is focused on graph sharing games: game T and game R. In this chapter we define them formally as particular cases of a more abstract setting, in which the game is played on an arbitrary set of weighted items (not necessarily the vertex set of a graph) and the rules of play are not specified explicitly. We also define constrained sharing games, which are very useful for the analysis of ordinary sharing games, and state some of their basic properties.

4.1. Games as rooted trees

The games we consider are two-player finite combinatorial games with perfect information. Informal specifications of such games usually involve some combinatorial rules of play and some way of evaluating the outcome of the game. We try to formalize these games so that the intuition standing behind two-player combinatorial games is not lost.

In our formalism, a *game* is a rooted tree with edges oriented naturally (so that every vertex is reachable from the root) and possibly with some additional data assigned to the edges. A vertex of the tree is a *position* in the game. The root is the *initial position*, and every leaf is a *final position*. A position q is a *successor* of a position p if there is an edge, called a *move* at p, leading from p to q. A position at an even or odd distance from the root is called *even* or *odd*, respectively. Thus the initial position is even.

The tree just defined represents the intuitive notion of a game as follows. The game starts at the initial position p_0. The first player has a choice to move to any position being a successor of p_0. Suppose it moves to a position p_1. Now, the second player can move to any position that is a successor of p_1. The situation repeats, alternating moves of the first and the second player, until a final position is reached, at which point the game is over. This way the players construct a path $p_0 \dots p_n$ in the tree leading from the initial position to a final one, which determines the outcome of the game.

Following this intuition, we say that an even position is *the first player's turn* and a move at an even position is *the first player's move*. Similarly, an odd position is *the second player's turn* and a move at an odd position is *the second player's move*. The *line of play* to a position p is the path in the tree from the initial position to p. Every line of play alternates moves of the first and the second player starting from the first player's one.

A *strategy* of the ith player in a game G is a subtree of G obtained by removing, at every non-final turn of the ith player, all but one moves, and removing all components of the resulting forest except the one containing the initial position. Therefore, a strategy determines completely the ith player's moves in the game while leaving all possible moves of the other player. The intersection of two strategies, one of the first and one of the second player, gives a single line of play leading to a final position.

We consider two kinds of games, which differ in the way of evaluating final positions. In *winner-loser games* the evaluation indicates, at every final position, which player *wins* and which *loses* the game. In *c-sum games*, where $c \in \mathbb{R}$, the evaluation assigns two real numbers to every final position, called the first and the second player's *values*, which sum up to c. The evaluations of final positions determine the evaluations of non-final positions by a bottom-up induction, in a way that expresses the players' *goals* in the game. In winner-loser games the obvious goal of each player is to win. Thus the ith player wins the game at a non-final position p when

- the ith player wins the game at some successor of p, if p is the ith player's turn,
- the ith player wins the game at every successor of p, if p is the other player's turn.

In c-sum games the goal of each player is to maximize its value. Thus the ith player's value at a non-final position p is

- the maximum of the ith player's values in the successors of p, if p is the ith player's turn,
- the minimum of the ith player's values in the successors of p, if p is the other player's turn.

Note that the players' values at every non-final position in a c-sum game sum up to c as well. The evaluation of the whole game is the evaluation of its initial position. A position is *optimal* for the ith player if its evaluation is at least as good for the ith player as the evaluation of the whole game. A move of the ith player is *optimal* if it leads to a position that is optimal for the ith player. A strategy S of the ith player is *optimal* if all positions in S are optimal for the ith player, or equivalently, if all moves of the ith player in S are optimal. A line of play is *optimal* if all its positions are optimal for both players. The intersection of two optimal strategies, one of the first and one of the second player, gives an optimal line of play.

The *subgame* of G at a position p, denoted by $\mathsf{G}(p)$, is the game represented by the rooted tree $\mathsf{G}(p)$, that is, the subtree of G rooted at p containing all positions of G reachable from p by an oriented path. In particular, the subgame of G at the initial position is the whole G. Intuitively, the subgame at p represents what can be played further after the game has reached the position p. The evaluation of $\mathsf{G}(p)$ is computed the same way as for G. However, the value of a particular final position f in $\mathsf{G}(p)$ may be different from its value in G, as the evaluation may take into account the entire line of play leading to f.

We usually call the first player Alice and the second player Bob. To avoid confusion, when we define a strategy in a game G in terms of strategies for other (auxiliary) games, we use the names Alice and Bob for the players in G, while calling the players the first and the second in the auxiliary games.

4.2. Abstract sharing games

Let I be a finite set of *items* equipped with a function $w : I \to [0, \infty)$ assigning a *weight* to each item. Intuitively, a *sharing game* on I is a combinatorial game in which moves consist in taking items from I one by one until no more items remain, and the goal of both players is to maximize the total weight of their collected items. More formally, it is a game G such that

- every move is labeled by one element of I; such a move is considered *taking* that item;
- for every position p the line of play leading to p consists of moves taking pairwise distinct items; an item taken with a move of the ith player on the line of play leading to p is called *taken* or *collected* by the ith player at the position p;
- for every position p the moves at p take pairwise distinct items;
- at every final position all items are taken; thus every line of play leading to a final position consists of exactly $|I|$ moves.

An item $x \in I$ is *available* at a position p if there is a move at p taking x. We denote the item set of G by $I(\mathsf{G})$ and define $w(\mathsf{G}) = w(I(\mathsf{G}))$.

The ith player's *partial outcome* at a position p (or the ith player's *outcome* at p when p is a final position) is evaluated as the total weight of the items taken by the ith player at the position p. The sum of both players' outcomes at every final position is $w(I)$, so G is a $w(I)$-sum game. The ith player's value at a final position is the outcome of the ith player at this position. The players' values at every other position in G and of the whole G are defined inductively so that the goal of each player is to maximize its own value. We denote the first and the second players' values of G by $\mathrm{val}_1\,\mathsf{G}$ and $\mathrm{val}_2\,\mathsf{G}$, respectively. We also say that the ith player can *secure* outcome $\mathrm{val}_i\,\mathsf{G}$ or that the ith player's *guaranteed outcome* is $\mathrm{val}_i\,\mathsf{G}$. Clearly, $\mathrm{val}_1\,\mathsf{G} + \mathrm{val}_2\,\mathsf{G} = w(\mathsf{G})$.

For a position p in G we define $\partial_{\mathsf{G}}(p) = b - a$, where a and b are respectively the first and the second players' partial outcomes in G at the position p. We write just $\partial(p)$ when it is clear from the context what game G is considered.

Sharing games $\mathsf{G}_1, \ldots, \mathsf{G}_k$ with pairwise disjoint item sets can be *added* to form a new game, the *sum* of $\mathsf{G}_1, \ldots, \mathsf{G}_k$. Intuitively, it is the game in which the players play simultaneously the games $\mathsf{G}_1, \ldots, \mathsf{G}_k$ making one move in one of $\mathsf{G}_1, \ldots, \mathsf{G}_k$ in each turn. Formally, it is a sharing game G with $I(\mathsf{G}) = I(\mathsf{G}_1) \cup \ldots \cup I(\mathsf{G}_k)$ and k *projection* functions $V(\mathsf{G}) \ni p \mapsto p^i \in V(\mathsf{G}_i)$ (for $1 \leqslant i \leqslant k$) such that

- if pq is a move in G with label $x \in I(\mathsf{G}_i)$ then $q^j = p^j$ for $j \neq i$ and $p^i q^i$ is a move in G_i with label x;
- if f is a final position in G then f^i is a final position in G_i for every i.

A straightforward induction shows that such a game G exists and is unique up to label-preserving isomorphism. We write $\mathsf{G}_1 + \ldots + \mathsf{G}_k$ to denote any game that is a sum of $\mathsf{G}_1, \ldots, \mathsf{G}_k$. With this notation, addition of sharing games is an associative and commutative operation.

4.3. Graph sharing games

Graph sharing games are sharing games played on the vertex set of a graph. We define them in terms of a *rule of play* expressed by some condition that every valid position in the game must satisfy. We implicitly assume that the game contains all possible moves except those leading to positions forbidden by the rule. Since we do not specify what precisely the vertices of the game tree are, such games are determined uniquely up to label-preserving isomorphism.

Let G be a weighted connected graph with weight function $w : V(G) \to [0, \infty)$. We define $\mathsf{T}(G)$ to be the sharing game with item set $V(G)$, weight function w, and the rule of play that requires the set of taken vertices at every position to be connected. Therefore, every vertex is available at the starting position, while at every other position p only those vertices are available that are adjacent to the set of taken vertices at p.

Now, let G be a weighted rooted graph with weight function $w : V(G) \to [0, \infty)$ and root set R such that every component of G contains a vertex from R. We define $\mathsf{Tr}(G)$ to be the sharing game with item set $V(G)$, weight function w, and the rule of play that requires every component of the subgraph of G induced on the set of taken vertices to contain a vertex from R. Therefore, a vertex v is available at a position p if $v \in R \cup N_G(T)$, where T is the set of taken vertices at p. When $C_1, \ldots, C_k$ are the components of G, we have $\mathsf{Tr}(G) = \mathsf{Tr}(C_1) + \ldots + \mathsf{Tr}(C_k)$.

The subgame of $\mathsf{T}(G)$ at a position p is the game $\mathsf{Tr}(G - T)$, where T is the set of taken vertices at p and $G - T$ is considered as a rooted graph with root set $N_G(T)$. The subgame of $\mathsf{Tr}(G)$ at a position p is the game $\mathsf{Tr}(G - T)$, where T is the set of taken vertices at p and $G - T$ is considered as a rooted graph with root set $R(G) \cup N_G(T)$.

Let G be a weighted connected graph with weight function $w : V(G) \to [0, \infty)$. We define $\mathsf{R}(G)$ to be the sharing game with item set $V(G)$, weight function w, and the rule of play that requires the set of untaken vertices at every position to be connected. In particular, when G is a weighted tree, a vertex v is available at a position p if v is a leaf of $G[U]$, where U is the set of untaken vertices at p. The subgame of $\mathsf{R}(G)$ at a position p is the game $\mathsf{R}(G[U])$, where U is the set of untaken vertices at p.

4.4. Constrained sharing games

Let G be a sharing game. For any $x, y \in \mathbb{R}$ with $y \leqslant 0$ we define a winner-loser game $\mathsf{G}(x, y)$ as follows. We take the game tree G. We remove all moves leading to odd positions p with $\partial(p) > x$ and even positions p with $\partial(p) < y$ together with all components of the resulting forest except the one containing the initial position. The game obtained this way (with the same initial position) is $\mathsf{G}(x, y)$. The winner is the first player at odd final positions and the second player at even final positions. That is, the player that cannot move loses the game.

It follows directly from the definition of $\mathsf{G}(x, y)$ that for every position p in $\mathsf{G}(x, y)$ we have

$$\mathsf{G}(x, y)(p) = \begin{cases} \mathsf{G}(p)\big(\partial(p) - y, \partial(p) - x\big) & \text{if } p \text{ is odd,} \\ \mathsf{G}(p)\big(x - \partial(p), y - \partial(p)\big) & \text{if } p \text{ is even.} \end{cases}$$

Furthermore, if $x' \geqslant x$, $y' \geqslant y$, and $\mathsf{G}(x, y)$ is won by the first player then the first player also wins $\mathsf{G}(x', y')$. For a line of play $p_0 \ldots p_n$ in $\mathsf{G}(x, y)$ the following two conditions are equivalent:

- $p_0 \ldots p_n$ is optimal for $\mathsf{G}(x, y)$,
- $\mathsf{G}(x, y)(p_i)$ is won by the same player as $\mathsf{G}(x, y)$ if i is even and by the other player if i is odd, for $0 \leqslant i \leqslant n$.

We define

$$\text{val}^\star \mathsf{G} = \min\{x \in \mathbb{R} : \mathsf{G}(x,0) \text{ is won by the first player}\},$$

where we adopt the convention that $\min \emptyset = \infty$. It follows from the definition of $\mathsf{G}(x,0)$ that the minimum above always exists. Moreover, $\text{val}\,\mathsf{G}^\star = \infty$ is possible only when $|I(\mathsf{G})|$ is even. Clearly,

$$\text{val}^\star \mathsf{G} \in \{w(B) - w(A) : A, B \subseteq I(\mathsf{G})\} \cup \{\infty\}.$$

Thus if $\text{val}^\star \mathsf{G} < \infty$ then $\text{val}^\star \mathsf{G} \leqslant w(\mathsf{G})$. For $|I(\mathsf{G})|$ odd we have more.

Proposition 4.1. *If* $|I(\mathsf{G})|$ *is odd then* $\text{val}^\star \mathsf{G} \leqslant \text{val}_2 \mathsf{G}$.

Proof. Let S be an optimal strategy of Alice in G. For every position p in S, since Bob's partial outcome at p is at most $\text{val}_2 \mathsf{G}$, we have $\partial(p) \leqslant \text{val}_2 \mathsf{G}$. This and the assumption that $|I(\mathsf{G})|$ is odd imply that no even position in S is final for $\mathsf{G}(\text{val}_2 \mathsf{G}, 0)$. Thus $\mathsf{S} \cap \mathsf{G}(\text{val}_2 \mathsf{G}, 0)$ is a winning strategy of Alice in $\mathsf{G}(\text{val}_2 \mathsf{G}, 0)$. This yields $\text{val}^\star \mathsf{G} \leqslant \text{val}_2 \mathsf{G}$. □

Proposition 4.2. *For any sharing games* $\mathsf{G}_1, \ldots, \mathsf{G}_k$ *with pairwise disjoint item sets we have*

$$\text{val}^\star(\mathsf{G}_1 + \ldots + \mathsf{G}_k) \geqslant \min_{i=1,\ldots,k} \text{val}^\star \mathsf{G}_i.$$

Proof. Let $\mathsf{G} = \mathsf{G}_1 + \ldots + \mathsf{G}_k$ and $x < \min_{i=1,\ldots,k} \text{val}^\star \mathsf{G}_i$. It follows that the second player wins the game $\mathsf{G}_i(x,0)$ for every i. We show that Bob wins the game $\mathsf{G}(x,0)$, providing him with a strategy to answer all possible moves of Alice.

Let S_i be a winning strategy of the second player in $\mathsf{G}_i(x,0)$. A winning strategy S of Bob in $\mathsf{G}(x,0)$ looks as follows: whenever Alice takes an item from $I(\mathsf{G}_i)$, he answers according to S_i. We need to show that Alice's move in $\mathsf{G}(x,0)$ is a valid move for $\mathsf{G}_i(x,0)$ and Bob's answer according to S_i in $\mathsf{G}_i(x,0)$ is a valid move in $\mathsf{G}(x,0)$.

For a position p in G let p^i denote the projection of p onto G_i. Let p be an even position in S and pq be a valid move of Alice in $\mathsf{G}(x,0)$ taking an item from $I(\mathsf{G}_i)$. Thus $p^i q^i$ is a move in G_i, while $q^j = p^j$ for $j \neq i$. Since each q_j with $j \neq i$ is a valid even position in $\mathsf{G}_j(x,0)$, we have

$$\partial(q^j) = \partial(p^j) \geqslant 0 \quad \text{for } j \neq i,$$
$$\partial(q^i) \leqslant \partial(q^i) + \sum_{j \neq i} \partial(q^j) = \partial(q) \leqslant x.$$

This shows that the move $p^i q^i$ is valid for $\mathsf{G}_i(x,0)$ and thus belongs to S_i. Bob wants to answer in $\mathsf{G}(x,0)$ with a move qr such that $q^i r^i$ is the second player's answer to the move $p^i q^i$ in S_i and $r^j = q^j = p^j$ for $j \neq i$. We have

$$\partial(r^j) = \partial(p^j) \geqslant 0 \quad \text{for } j \neq i,$$
$$\partial(r) = \partial(r^i) + \sum_{j \neq i} \partial(r^j) \geqslant 0,$$

which shows that qr is a valid move in $\mathsf{G}(x,0)$. □

The following result is a joint work with Piotr Micek and appears in [14] for the special case that $\mathsf{G}_1, \ldots, \mathsf{G}_k$ are games Tr on graphs.

Theorem 4.3. *For any sharing games* $\mathsf{G}_1, \ldots, \mathsf{G}_k$ *with pairwise disjoint item sets and* $|I(\mathsf{G}_1 + \ldots + \mathsf{G}_k)|$ *odd we have*

$$\operatorname{val}_1(\mathsf{G}_1 + \ldots + \mathsf{G}_k) \geqslant \tfrac{1}{2}\big(w(\mathsf{G}_1 + \ldots + \mathsf{G}_k) - \max_{i=1,\ldots,k} \max_{\mathsf{H}\in\mathcal{H}_i} \operatorname{val}^\star \mathsf{H}\big),$$

where $\mathcal{H}_i$ *denotes the family of subgames of* G_i *with item sets of odd size.*

Proof. Let $\mathsf{G} = \mathsf{G}_1 + \ldots + \mathsf{G}_k$. The argument goes by induction on $|I(\mathsf{G})|$. Define

$$d = \max_{i=1,\ldots,k} \max_{\mathsf{H}\in\mathcal{H}_i} \operatorname{val}^\star \mathsf{H}.$$

We show that Alice has a strategy S in G such that $\partial(f) \leqslant d$ for every final position f in S, which is equivalent to the conclusion of the theorem.

To describe S we simulate the game and define Alice's answers to all possible moves of Bob. Our simulation stops at either an odd final position f with $\partial(f) \leqslant d$ or an even position p with $\partial(p) \leqslant 0$. In the latter case, by the induction hypothesis applied to $\mathsf{G}(p)$, there is a strategy S' for the first player in $\mathsf{G}(p)$ such that $\partial_{\mathsf{S}'}(f) \leqslant d$ for every final position f in S'. It follows that for every final position f in S' we have $\partial_{\mathsf{S}}(f) = \partial_{\mathsf{S}}(p) + \partial_{\mathsf{S}'}(f) \leqslant d$. Thus Alice can continue the game according to the strategy S'.

Assume without loss of generality that

$$\min_{i=1,\ldots,k} \operatorname{val}^\star \mathsf{G}_i = \operatorname{val}^\star \mathsf{G}_1. \tag{4.1}$$

At least one of $|I(\mathsf{G}_1)|, \ldots, |I(\mathsf{G}_k)|$ is odd, so $\operatorname{val}^\star \mathsf{G}_1 < \infty$. Define

$$X = \big\{w(B) - w(A) : A, B \subseteq I(\mathsf{G})\big\},$$
$$\varepsilon = \min\big\{y - x : x, y \in X \text{ and } x < y\big\}.$$

Let S_1 be a winning strategy of the first player in $\mathsf{G}_1(\operatorname{val}^\star \mathsf{G}_1, 0)$. For $i \neq 1$ let S_i be a winning strategy of the second player in $\mathsf{G}_i(\operatorname{val}^\star \mathsf{G}_i - \varepsilon, 0)$. For a position p in G let p^i denote the projection of p onto G_i.

We build the strategy S on top of $\mathsf{S}_1, \ldots, \mathsf{S}_k$. In our description of the strategy, every move in G taking an item from $I(\mathsf{G}_i)$ is interpreted as a move in G_i. Alice acts as the first player in G_1 and the second player in every other G_i. She starts in G_1 according to S_1. Whenever Bob makes a move in G_1 that is valid for $\mathsf{G}_1(\operatorname{val}^\star \mathsf{G}_1, 0)$, Alice answers in G_1 according to S_1. Whenever Bob makes a move in G_i that is valid for $\mathsf{G}_i(\operatorname{val}^\star \mathsf{G}_i - \varepsilon, 0)$ with $i \neq 1$, Alice answers there following S_i. Every odd position p in G that is reached according to this strategy satisfies $\partial(p^1) \leqslant \operatorname{val}^\star \mathsf{G}_1$ and $\partial(p^i) \geqslant 0$ for $i \neq 1$. The presented strategy can no longer be continued after one of the following situations occurs:

(a) Bob makes a move in G_1 not valid for the second player in $\mathsf{G}_1(\operatorname{val}^\star \mathsf{G}_1, 0)$;
(b) Bob makes a move in G_j with $j \neq 1$ that is not valid for the first player in $\mathsf{G}_j(\operatorname{val}^\star \mathsf{G}_j - \varepsilon, 0)$;
(c) Alice's last move has reached a final position f.

We show that if (a) or (b) happens then an even position q in G with $\partial(q) \leqslant 0$ has been reached, while if (c) happens then we have $\partial(f) \leqslant d$.

For the cases (a) and (b), let pq be the last move of Bob in G. In the case (a) we have $\partial(q^1) < 0$ and $q^i = p^i$ for $i \neq 1$, and thus

$$\partial(q) = \partial(q^1) - \sum_{i\neq 1} \partial(q^i) = \partial(q^1) - \sum_{i\neq 1} \partial(p^i) < 0.$$

In the case (b) we have $\partial(q^j) > \text{val}^\star \mathsf{G}_j - \varepsilon$. This implies $\partial(q^j) \geqslant \text{val}^\star \mathsf{G}_j$ as $\text{val}^\star \mathsf{G}_j \in X$. We also have $q^i = p^i$ for $i \neq j$. Thus

$$\begin{aligned}\partial(q) &= \partial(q^1) - \sum_{i\neq 1} \partial(q^i)\\ &= \partial(p^1) - \partial(q^j) - \sum_{i\neq 1,j} \partial(p^i)\\ &\leqslant \text{val}^\star \mathsf{G}_1 - \text{val}^\star \mathsf{G}_j \leqslant 0.\end{aligned}$$

Finally, in the case (c) we have

$$\partial(f) = \partial(f^1) - \sum_{i\neq 1} \partial(f^i) \leqslant \text{val}^\star \mathsf{G}_1 \leqslant d. \qquad \square$$

Corollary 4.4. *For any sharing games* $\mathsf{G}_1, \ldots, \mathsf{G}_k$ *with pairwise disjoint item sets and* $|I(\mathsf{G}_1 + \ldots + \mathsf{G}_k)|$ *odd we have*

$$\text{val}_1(\mathsf{G}_1 + \ldots + \mathsf{G}_k) \geqslant \tfrac{1}{2}\big(w(\mathsf{G}_1 + \ldots + \mathsf{G}_k) - \max_{i=1,\ldots,k} w(\mathsf{G}_i)\big).$$

Proof. The conclusion follows directly from Theorem 4.3 and the fact that $\text{val}^\star \mathsf{G}_i \leqslant w(\mathsf{G}_i)$ for $|I(\mathsf{G}_i)|$ odd. $\square$

Chapter 5

Game T on odd trees

Now, we show that in game T on a weighted tree with an odd number of vertices Alice can secure at least $\frac{1}{4}$ of the total weight. We start with a simpler proof giving at least $\frac{1}{6}$ of the total weight. The results are a joint work with Piotr Micek and are contained in [14].

Proposition 5.1. *For every weighted tree T with an odd number of vertices Alice has a strategy in $\mathsf{T}(T)$ to collect vertices of total weight at least $\frac{1}{6}w(T)$.*

Proof. Let T be a weighted tree with an odd number of vertices. If T has only one vertex, the conclusion holds trivially. Thus assume that T has at least three vertices. If any vertex of T has weight at least $\frac{1}{6}w(T)$, Alice takes this vertex with her first move and thus guarantees herself final gain at least $\frac{1}{6}w(T)$. Thus suppose that every vertex of T has weight less than $\frac{1}{6}w(T)$.

Alice's strategy is as follows. She starts by taking a weighted center v of T, which exists by Proposition 3.1. Bob answers by taking some vertex b. Now, let $C_1, \ldots, C_k$ be the components of $T - \{v, b\}$ considered as rooted trees with roots being the vertices adjacent to $\{v, b\}$. Since v is a weighted center, every C_i has weight at most $\frac{1}{2}w(T)$. The subgame of $\mathsf{T}(T)$ at the current position is $\mathsf{Tr}(C_1) + \ldots + \mathsf{Tr}(C_k)$. By Corollary 4.4 Alice has a strategy in this subgame to collect vertices of total weight at least

$$\begin{aligned}\tfrac{1}{2}\big(w(T - \{v, b\}) - \max_{i=1,\ldots,k} w(C_k)\big) &\geqslant \tfrac{1}{2}\big(w(T) - w(v) - w(b) - \tfrac{1}{2}w(T)\big)\\ &> \tfrac{1}{2}\big(\tfrac{1}{3}w(T) - w(v)\big)\\ &\geqslant \tfrac{1}{6}w(T) - w(v).\end{aligned}$$

Since she has also taken v, she gathers more than $\frac{1}{6}w(T)$ in total. □

Theorem 5.2. *For every weighted tree T with an odd number of vertices Alice has a strategy in $\mathsf{T}(T)$ to collect vertices of total weight at least $\frac{1}{4}w(T)$.*

Proof. We prove that for a suitable constant c Alice can secure at least $c \cdot w(T)$ in $\mathsf{T}(T)$ for any weighted tree T with an odd number of vertices. The proof goes by induction on the number of vertices of T. To get through the induction step we bound the constant c from above. The greatest c for which the argument works turns out to be $\frac{1}{4}$.

Let T be a weighted tree with an odd number of vertices. A *rooted subtree* of T is any proper subtree R of T with $T - R$ connected and with the root defined to be the vertex adjacent to $T - R$. For a rooted subtree R of T let $\mathsf{S}_1(R)$ and $\mathsf{S}_2(R)$ denote respectively optimal strategies of the first and the second player in $\mathsf{Tr}(R)$. That is, the strategy $\mathsf{S}_i(R)$ allows the ith player to collect vertices of total weight at least $\mathrm{val}_i\, \mathsf{Tr}(R)$ in $\mathsf{Tr}(R)$.

Suppose first that there is a rooted subtree T_0 of T with an even number of vertices and with $\mathrm{val}_2\, \mathsf{Tr}(T_0) \geqslant c \cdot w(T_0)$. Since $T - T_0$ has an odd number of vertices, by the induction hypothesis Alice has a strategy S' in $\mathsf{T}(T - T_0)$ to collect vertices of total weight at least $c \cdot w(T - T_0)$. We now construct a strategy for Alice in $\mathsf{T}(T)$. She starts in $T - T_0$ according to the strategy S'. Whenever Bob plays in $T - T_0$, Alice responds in $T - T_0$ following S'. Whenever Bob plays in T_0, Alice responds in T_0 according to the strategy $\mathsf{S}_2(T_0)$. The parities of T_0 and $T - T_0$ are so chosen that Alice makes the last move in both parts. This way Alice's total gain on T is at least

$$c \cdot w(T - T_0) + \mathrm{val}_2\, \mathsf{Tr}(T_0) \geqslant c \cdot w(T - T_0) + c \cdot w(T_0) = c \cdot w(T),$$

which completes the proof for the considered case. Thus suppose for the rest of the proof that every rooted subtree T_0 of T with an even number of vertices satisfies

$$\mathrm{val}_2\, \mathsf{Tr}(T_0) < c \cdot w(T_0). \tag{5.1}$$

Suppose that there is a rooted subtree T_1 of T with an odd number of vertices such that $w(T_1) \leqslant \frac{1}{2}w(T)$ and $\mathrm{val}_2\, \mathsf{Tr}(T_1) \geqslant c \cdot w(T)$. Let T_0 denote the rooted subtree $T - T_1$. Thus

$$w(T_0) \geqslant \tfrac{1}{2}w(T). \tag{5.2}$$

The strategy for Alice on T is as follows. She starts in T_0 according to $\mathsf{S}_1(T_0)$, that is, taking the root of T_0. Whenever Bob plays in T_0, Alice answers in T_0 following $\mathsf{S}_1(T_0)$. Whenever Bob plays in T_1, Alice answers in T_1 according to $\mathsf{S}_2(T_1)$. Alice continues this strategy until T_0 or T_1 runs out of vertices. We claim that already at that moment Alice has collected vertices of total weight at least $c \cdot w(T)$.

If T_1 has been entirely taken first then Alice, realizing the whole strategy $\mathsf{S}_2(T_1)$, has secured at least $\mathrm{val}_2\, \mathsf{Tr}(T_1) \geqslant c \cdot w(T)$. Thus suppose that the whole T_0 has been taken before T_1. It follows that Alice, realizing the whole strategy $\mathsf{S}_1(T_0)$, has taken from T_0 at least

$$\begin{aligned} \mathrm{val}_1\, \mathsf{Tr}(T_0) &> (1-c)w(T_0) && \text{by (5.1)} \\ &\geqslant \tfrac{1}{2}(1-c)w(T) && \text{by (5.2)} \\ &\geqslant c \cdot w(T), \end{aligned}$$

where the last inequality holds for $c \leqslant \frac{1}{3}$.

Now, suppose that every rooted subtree T_1 of T with an odd number of vertices and with $w(T_1) \leqslant \frac{1}{2}w(T)$ satisfies $\mathrm{val}_2\, \mathsf{Tr}(T_1) < c \cdot w(T)$, which by Proposition 4.1 implies

$$\mathrm{val}^\star\, \mathsf{Tr}(T_1) \leqslant c \cdot w(T). \tag{5.3}$$

Let v be a weighted center of T. Alice starts by taking v. Bob responds by taking some vertex b. Now, let $C_1, \ldots, C_k$ be the components of $T - \{v, b\}$ considered as rooted trees. The subgame of $\mathsf{T}(T)$ at the current position is $\mathsf{Tr}(C_1) + \ldots + \mathsf{Tr}(C_k)$. Since every C_i has weight at most $\frac{1}{2}w(T)$, we have (5.3) for every rooted subtree T_1 of C_i with an odd number of vertices. Thus by Theorem 4.3 Alice has a strategy in $\mathsf{Tr}(C_1) + \ldots + \mathsf{Tr}(C_k)$ to collect vertices of total weight at least

$$\tfrac{1}{2}\big(w(T - \{v, b\}) - c \cdot w(T)\big) \geqslant \tfrac{1}{2}\big((1-c)w(T) - w(b)\big) - w(v).$$

Therefore, her total gain on T is at least $\frac{1}{2}\big((1-c)w(T) - w(b)\big)$. A complementary strategy for Alice is to start with b and do anything afterwards. The better of these two strategies gives Alice at least

$$\max\big\{\tfrac{1}{2}\big((1-c)w(T) - w(b)\big), w(b)\big\} \geqslant \tfrac{1}{3}(1-c)w(T) \geqslant c \cdot w(T),$$

where the last inequality holds for $c \leqslant \frac{1}{4}$. □

Chapter 6

Optimal strategies in game T on trees

In this chapter we provide strategies for both players in the game $\mathsf{G} = \mathsf{Tr}(F)$ played on a weighted rooted forest F, which are optimal for G and also for every constrained game $\mathsf{G}(x, y)$ with $y \leqslant 0$. Then, we use these strategies to construct optimal strategies for both players in game T on a weighted tree. The strategies we present are computable in polynomial time in the sense that an optimal move at any given position can be computed in polynomial time.

6.1. Properties of optimal lines of play

Recall that for any position p in $\mathsf{G}(x, y)$ we have

$$\mathsf{G}(x, y)(p) = \begin{cases} \mathsf{G}(p)\big(\partial(p) - y, \partial(p) - x\big) & \text{if } p \text{ is odd,} \\ \mathsf{G}(p)\big(x - \partial(p), y - \partial(p)\big) & \text{if } p \text{ is even.} \end{cases}$$

Recall also that if $x' \geqslant x$, $y' \geqslant y$, and $\mathsf{G}(x, y)$ is won by the first player then the first player wins $\mathsf{G}(x', y')$ as well. In the following we use these observations without explicit reference.

For convenience, we say that a line of play $p_0 \dots p_n$ in G is optimal for $\mathsf{G}(x, y)$ when it is an optimal line of play in $\mathsf{G}(x, y)$ under the usual definition or some prefix $p_0 \dots p_k$ of it with p_k being a final position in $\mathsf{G}(x, y)$ is an optimal line of play in $\mathsf{G}(x, y)$.

Lemma 6.1. *Let G be a sharing game. Let $p_0 \dots p_n$ be a line of play in G optimal for every $\mathsf{G}(x, y)$ and such that*

$$\operatorname{val}^\star \mathsf{G}(p_i) \leqslant \operatorname{val}^\star \mathsf{G} \quad \textit{for } 0 \leqslant i < n.$$
$$\operatorname{val}^\star \mathsf{G}(p_n) \geqslant \operatorname{val}^\star \mathsf{G}.$$

Let $s_i = \partial(p_i)$. We have

$$\begin{aligned} & s_i \leqslant \operatorname{val}^\star \mathsf{G} && \textit{for } 1 \leqslant i \leqslant n \textit{ and } i \textit{ odd,} \\ & s_i \geqslant \operatorname{val}^\star \mathsf{G}(p_i) && \textit{for } 1 \leqslant i < n \textit{ and } i \textit{ odd,} \\ & s_i \geqslant 0 && \textit{for } 0 \leqslant i \leqslant n \textit{ and } i \textit{ even,} \\ & s_i \leqslant \operatorname{val}^\star \mathsf{G} - \operatorname{val}^\star \mathsf{G}(p_i) && \textit{for } 0 \leqslant i \leqslant n \textit{ and } i \textit{ even, if } \operatorname{val}^\star \mathsf{G} < \infty. \end{aligned}$$

Moreover, we have

$$\begin{aligned} & s_n = \operatorname{val}^\star \mathsf{G} < \operatorname{val}^\star \mathsf{G}(p_n) && \textit{if } n \textit{ is odd,} \\ & s_n = \operatorname{val}^\star \mathsf{G} - \operatorname{val}^\star \mathsf{G}(p_n) = 0 && \textit{if } n \textit{ is even and } \operatorname{val}^\star \mathsf{G} < \infty. \end{aligned}$$

Proof. We prove the following statements:

$$
\begin{array}{lll}
 & s_i \leqslant \mathrm{val}^\star \mathsf{G} & \text{for } 1 \leqslant i \leqslant n \text{ and } i \text{ odd,} \\
 & s_i \geqslant 0 & \text{for } 0 \leqslant i \leqslant n \text{ and } i \text{ even,} \\
(6.1) & s_i \leqslant \mathrm{val}^\star \mathsf{G} - \mathrm{val}^\star \mathsf{G}(p_i) & \text{for } 0 \leqslant i \leqslant n \text{ and } i \text{ even, if } \mathrm{val}^\star \mathsf{G} < \infty, \\
 & s_i < \mathrm{val}^\star \mathsf{G} \quad \text{with } 1 \leqslant i \leqslant n \text{ and } i \text{ odd implies} & s_i \geqslant \mathrm{val}^\star \mathsf{G}(p_i), \\
 & s_i = \mathrm{val}^\star \mathsf{G} \quad \text{with } 1 \leqslant i \leqslant n \text{ and } i \text{ odd implies} & s_i < \mathrm{val}^\star \mathsf{G}(p_i).
\end{array}
$$

They are enough for the conclusion of the lemma. Indeed, for $1 \leqslant i \leqslant n$ and i odd, $s_i < \mathrm{val}^\star \mathsf{G}$ implies $\mathrm{val}^\star \mathsf{G} > \mathrm{val}^\star \mathsf{G}(p_i)$, and $s_i = \mathrm{val}^\star \mathsf{G}$ implies $\mathrm{val}^\star \mathsf{G} < \mathrm{val}^\star \mathsf{G}(p_i)$. This and the assumptions of the lemma show that if $1 \leqslant i < n$ and i is odd then $s_i < \mathrm{val}^\star \mathsf{G}$, which yields $s_i \geqslant \mathrm{val}^\star \mathsf{G}(p_i)$, while if n is odd then $s_n = \mathrm{val}^\star \mathsf{G} < \mathrm{val}^\star \mathsf{G}(p_n)$. That $s_n = \mathrm{val}^\star \mathsf{G} - \mathrm{val}^\star \mathsf{G}(p_n) = 0$ if n is even and $\mathrm{val}^\star \mathsf{G} < \infty$ follows directly from the inequalities $\mathrm{val}^\star \mathsf{G}(p_n) \geqslant \mathrm{val}^\star \mathsf{G}$ and $0 \leqslant s_n \leqslant \mathrm{val}^\star \mathsf{G} - \mathrm{val}^\star \mathsf{G}(p_n)$.

We prove the statements (6.1) by induction: for $0 \leqslant k \leqslant n$, we assume that they hold for $0 \leqslant i < k$ and prove them for $i = k$.

Suppose k is even. For $1 \leqslant i < k$ and i odd, by the induction hypothesis, we have $s_i \leqslant \mathrm{val}^\star \mathsf{G}$ and we have $s_i < \mathrm{val}^\star \mathsf{G}(p_i)$ if $s_i = \mathrm{val}^\star \mathsf{G}$. This and the assumption that $\mathrm{val}^\star \mathsf{G}(p_i) \leqslant \mathrm{val}^\star \mathsf{G}$ imply $s_i < \mathrm{val}^\star \mathsf{G}$. Let

$$s' = \max\{s_i : 1 \leqslant i < k \text{ and } i \text{ odd}\}.$$

It follows that $s' < \mathrm{val}^\star \mathsf{G}$ and thus the second player wins $\mathsf{G}(s', 0)$. Since $p_0 \dots p_n$ is an optimal line of play for $\mathsf{G}(s', 0)$ and all moves up to the position p_{k-1} are valid for $\mathsf{G}(s', 0)$, the second player's move to the position p_k has to be valid as well. Thus $s_k \geqslant 0$. Suppose additionally that $\mathrm{val}^\star \mathsf{G} < \infty$. Since the first player wins $\mathsf{G}(\mathrm{val}^\star \mathsf{G}, 0)$, the game $\mathsf{G}(p_k)(\mathrm{val}^\star \mathsf{G} - s_k, -s_k)$ and thus $\mathsf{G}(p_k)(\mathrm{val}^\star \mathsf{G} - s_k, 0)$ are also won by the first player. This implies $\mathrm{val}^\star \mathsf{G}(p_k) \leqslant \mathrm{val}^\star \mathsf{G} - s_k$.

Now, suppose k is odd. Since $p_0 \dots p_n$ is an optimal line of play for $\mathsf{G}(\mathrm{val}^\star \mathsf{G}, 0)$ won by the first player, his move to position p_k has to be valid. Thus $s_k \leqslant \mathrm{val}^\star \mathsf{G}$.

Suppose $s_k = \mathrm{val}^\star \mathsf{G}$. The first player wins $\mathsf{G}(s_k, 0)$. Since this game played along $p_0 \dots p_n$ does not end before p_k, the game $\mathsf{G}(s_k, 0)(p_k)$ is won by the second player. Hence $\mathrm{val}^\star \mathsf{G}(p_k) > s_k$.

Now, suppose $s_k < \mathrm{val}^\star \mathsf{G}$. Let

$$s' = \max\{s_i : 1 \leqslant i \leqslant k \text{ and } i \text{ odd}\}.$$

As before, it follows that $s' < \mathrm{val}^\star \mathsf{G}$ and thus the second player wins $\mathsf{G}(s', 0)$. This game played along $p_0 \dots p_n$ does not end before p_k. Thus the first player wins $\mathsf{G}(p_k)(s_k, s_k - s')$. It follows that $\mathsf{G}(p_k)(s_k, 0)$ is won by the first player as well. Hence $\mathrm{val}^\star \mathsf{G}(p_k) \leqslant s_k$. □

Let F be a weighted rooted forest and p be a position in $\mathsf{Tr}(F)$. We define

$$\hat{w}(p) = \min\{\mathrm{val}^\star \mathsf{Tr}(T_1), \dots, \mathrm{val}^\star \mathsf{Tr}(T_k)\},$$

where $T_1, \dots, T_k$ is the collection of rooted trees (rooted forest) induced from F on the set of untaken vertices at position p. By Proposition 4.2 we have

$$\mathrm{val}^\star \mathsf{G}(p) \geqslant \hat{w}(p). \tag{6.2}$$

If T_i is a tree realizing the minimum in the definition of $\hat{w}(p)$ then the move at position p taking the root of T_i is *$\star$-minimal*. A line of play in $\mathsf{Tr}(F)$ is *$\star$-minimal* if all moves it consists of are $\star$-minimal.

Lemma 6.2. *Let* G *be a sharing game of the form* $\mathsf{G} = \mathsf{Tr}(F)$ *for a weighted rooted forest* F*. Let* $p_0 \dots p_n$ *be a* $\star$*-minimal line of play in* G *such that*

$$\begin{aligned} \hat{w}(p_i) &< \hat{w}(p_0) \quad \text{for } 1 \leqslant i < n, \\ \hat{w}(p_n) &\geqslant \hat{w}(p_0). \end{aligned}$$

Suppose that for $0 \leqslant i < n$ *the move* $p_i p_{i+1}$ *is optimal for every* $\mathsf{G}(p_i)(x, y)$*. Let* $s_i = \partial(p_i)$*. We have*

$$\begin{aligned} s_i &\leqslant \hat{w}(p_0) && \text{for } 1 \leqslant i \leqslant n \text{ and } i \text{ odd}, \\ s_i &\geqslant \hat{w}(p_i) && \text{for } 1 \leqslant i < n \text{ and } i \text{ odd}, \\ s_i &\geqslant 0 && \text{for } 0 \leqslant i \leqslant n \text{ and } i \text{ even}, \\ s_i &\leqslant \hat{w}(p_0) - \hat{w}(p_i) && \text{for } 0 \leqslant i < n \text{ and } i \text{ even, if } \hat{w}(p_0) < \infty. \end{aligned}$$

Moreover, we have

$$\begin{aligned} s_n &= \hat{w}(p_0) && \text{if } n \text{ is odd}, \\ s_n &= \hat{w}(p_0) - \hat{w}(p_n) = 0 && \text{if } n \text{ is even and } \hat{w}(p_0) < \infty. \end{aligned}$$

Proof. The proof goes by induction on $|V(F)|$. First suppose that the forest F consists of at least two trees $T_1, \dots, T_k$ and T_1 is the tree where the move $p_0 p_1$ is made. By $\star$-minimality all moves on $p_0 \dots p_n$ are made in T_1. Let p'_i denote the projection of p_i onto T_1. Clearly, $\hat{w}(p'_i) < \hat{w}(p'_0)$ for $1 \leqslant i < n$ and $\hat{w}(p'_n) \geqslant \hat{w}(p'_0)$, and thus we can apply the lemma inductively to $\mathsf{Tr}(T_1)$ and $p'_0 \dots p'_n$. This directly yields the conclusion of the lemma for G and $p_0 \dots p_n$.

It remains to prove the lemma for F being a single tree. In this case we have $\text{val}^\star \mathsf{G} = \hat{w}(p_0)$. If $\text{val}^\star \mathsf{G}(p_i) \leqslant \hat{w}(p_0)$ for every i with $1 \leqslant i < n$, then the conclusion follows directly from Lemma 6.1 and (6.2). Thus we assume this is not the case. Let k be the smallest index with $1 \leqslant k < n$ and $\text{val}^\star \mathsf{G}(p_k) > \hat{w}(p_0)$. We apply Lemma 6.1 to G and $p_0 \dots p_k$. This, the fact that $\text{val}^\star \mathsf{G} = \hat{w}(p_0)$, and (6.2) yield

$$\begin{aligned} s_i &\leqslant \hat{w}(p_0) && \text{for } 1 \leqslant i \leqslant k \text{ and } i \text{ odd}, \\ s_i &\geqslant \hat{w}(p_i) && \text{for } 1 \leqslant i < k \text{ and } i \text{ odd}, \\ s_i &\geqslant 0 && \text{for } 0 \leqslant i \leqslant k \text{ and } i \text{ even}, \\ s_i &\leqslant \hat{w}(p_0) - \hat{w}(p_i) && \text{for } 0 \leqslant i < k \text{ and } i \text{ even}. \end{aligned}$$

The above gives the conclusion for $0 \leqslant i < k$. Moreover, since $\text{val}^\star \mathsf{G}(p_k) > \hat{w}(p_0)$, Lemma 6.1 tells us that k is odd and

$$s_k = \hat{w}(p_0). \tag{6.3}$$

We construct a sequence $k_0, \dots, k_m$ of indices so that $k_0 = k$, $k_m = n$, and k_{i+1} is the smallest index such that $k_i < k_{i+1} \leqslant n$ and $\hat{w}(p_{k_{i+1}}) \geqslant \hat{w}(p_{k_i})$.

We prove the following by induction on r:

$$\begin{aligned} (-1)^k(s_i - s_k) &\leqslant \hat{w}(p_0) && \text{for } k \leqslant i \leqslant k_r \text{ and } i - k \text{ odd}, \\ (-1)^k(s_i - s_k) &\geqslant \hat{w}(p_i) && \text{for } k \leqslant i < k_r \text{ and } i - k \text{ odd}, \\ (-1)^k(s_i - s_k) &\geqslant 0 && \text{for } k \leqslant i \leqslant k_r \text{ and } i - k \text{ even}, \\ (-1)^k(s_i - s_k) &\leqslant \hat{w}(p_0) - \hat{w}(p_i) && \text{for } k \leqslant i < k_r \text{ and } i - k \text{ even}. \end{aligned} \tag{6.4}$$

$$(6.5)\qquad (-1)^k(s_{k_r}-s_k)=\hat{w}(p_{k_r}) \quad \text{if } k_r-k \text{ is odd,}$$
$$(6.6)\qquad (-1)^k(s_{k_r}-s_k)=0 \quad \text{if } k_r-k \text{ is even.}$$

We also show that if $k_r - k$ is odd then $k_r < n$. All this clearly holds for $r = 0$. For the inductive step we assume (6.5) and (6.6) and prove

$$(6.7)\qquad \begin{aligned} &(-1)^k(s_i-s_k)\leqslant \hat{w}(p_0) && \text{for } k\leqslant i\leqslant k_{r+1} \text{ and } i-k \text{ odd,}\\ &(-1)^k(s_i-s_k)\geqslant \hat{w}(p_i) && \text{for } k\leqslant i< k_{r+1} \text{ and } i-k \text{ odd,}\\ &(-1)^k(s_i-s_k)\geqslant 0 && \text{for } k\leqslant i\leqslant k_{r+1} \text{ and } i-k \text{ even,}\\ &(-1)^k(s_i-s_k)\leqslant \hat{w}(p_0)-\hat{w}(p_i) && \text{for } k\leqslant i< k_{r+1} \text{ and } i-k \text{ even.}\end{aligned}$$

$$(6.8)\qquad (-1)^k(s_{k_{r+1}}-s_k)=\hat{w}(p_{k_r})=\hat{w}(p_{k_{r+1}}) \quad \text{if } k_{r+1}-k \text{ is odd,}$$
$$(6.9)\qquad (-1)^k(s_{k_{r+1}}-s_k)=0 \quad \text{if } k_{r+1}-k \text{ is even.}$$

Once this is settled, we obtain $k_{r+1} < n$ for $k_{r+1} - k$ odd by (6.8) and the fact that $\hat{w}(p_{k_r}) < \hat{w}(p_0) \leqslant \hat{w}(p_n)$. By the lemma applied inductively to $\mathsf{G}(p_{k_r})$ and $p_{k_r} \dots p_{k_{r+1}}$ we have

$$\begin{aligned} &(-1)^{k_r}(s_i-s_{k_r})\leqslant \hat{w}(p_{k_r}) && \text{for } k_r< i\leqslant k_{r+1} \text{ and } i-k_r \text{ odd,}\\ &(-1)^{k_r}(s_i-s_{k_r})\geqslant \hat{w}(p_i) && \text{for } k_r< i< k_{r+1} \text{ and } i-k_r \text{ odd,}\\ &(-1)^{k_r}(s_i-s_{k_r})\geqslant 0 && \text{for } k_r\leqslant i\leqslant k_{r+1} \text{ and } i-k_r \text{ even,}\\ &(-1)^{k_r}(s_i-s_{k_r})\leqslant \hat{w}(p_{k_r})-\hat{w}(p_i) && \text{for } k_r\leqslant i< k_{r+1} \text{ and } i-k_r \text{ even,}\end{aligned}$$

which together with (6.5) and (6.6) yield (6.7), and we also have

$$(6.10)\qquad (-1)^{k_r}(s_{k_{r+1}}-s_{k_r})=\hat{w}(p_{k_r}) \quad \text{if } k_{r+1}-k_r \text{ is odd,}$$
$$(6.11)\qquad (-1)^{k_r}(s_{k_{r+1}}-s_{k_r})=\hat{w}(p_{k_r})-\hat{w}(p_{k_{r+1}})=0 \quad \text{if } k_{r+1}-k_r \text{ is even.}$$

Suppose that $k_{r+1} - k$ is even. If $k_r - k$ is odd then (6.5) and (6.10) imply (6.9). If $k_r - k$ is even then (6.9) follows from (6.6) and (6.11). Now, suppose that $k_{r+1} - k$ is odd. If $k_r - k$ is odd then (6.5) and (6.11) imply (6.8). Thus suppose $k_r - k$ is even. By (6.6) and (6.10) we have

$$(6.12)\qquad (-1)^k(s_{k_{r+1}}-s_k)=\hat{w}(p_{k_r}).$$

Since $\operatorname{val}^\star \mathsf{G}(p_k) > \hat{w}(p_0)$, the game $\mathsf{G}(p_k)(\hat{w}(p_0), 0)$ is won by the second player. It follows from (6.7) that this game does not end before $p_{k_{r+1}}$. This and (6.12) imply that $\mathsf{G}(p_{k_{r+1}})(\hat{w}(p_{k_r}), \hat{w}(p_{k_r}) - \hat{w}(p_0))$ is won by the first player and thus $\mathsf{G}(p_{k_{r+1}})(\hat{w}(p_{k_r}), 0)$ is also won by the first player. Hence

$$\operatorname{val}^\star \mathsf{G}(p_{k_{r+1}}) \leqslant \hat{w}(p_{k_r}).$$

On the other hand, by (6.2) we have

$$\operatorname{val}^\star \mathsf{G}(p_{k_{r+1}}) \geqslant \hat{w}(p_{k_{r+1}}) \geqslant \hat{w}(p_{k_r}).$$

The three inequalities above show that $\hat{w}(p_{k_r}) = \hat{w}(p_{k_{r+1}})$, which together with (6.12) gives (6.8). This completes the induction.

Now, by (6.3), (6.4), (6.6), and the fact that $n-k$ is even, we obtain the conclusion of the lemma for $k \leqslant i \leqslant n$. □

6.2. Optimality of $\star$-minimal lines of play

Theorem 6.3. *Let F be a weighted rooted forest. Every $\star$-minimal line of play in $\mathsf{Tr}(F)$ is optimal for $\mathsf{Tr}(F)$ and for every $\mathsf{Tr}(F)(x,y)$.*

Proof. The whole proof goes by induction on $|V(F)|$. It is enough to prove that every $\star$-minimal first move in $\mathsf{Tr}(F)$ is optimal for $\mathsf{Tr}(F)$ and for every $\mathsf{Tr}(F)(x,y)$, as the optimality of subsequent $\star$-minimal moves follows from the induction hypothesis applied to the subgame at the position after the first move. If F is a single rooted tree then there is only one starting move, which must be optimal. Thus for the remainder of the proof we assume that F consists of at least two rooted trees, one of which is T and the others form a forest U, that is, $F = T \cup U$. We prove the following statements:

- If the starting move taking the root of T is optimal for $\mathsf{Tr}(F)$ then every $\star$-minimal starting move on F is optimal for $\mathsf{Tr}(F)$.
- If the starting move taking the root of T is optimal for $\mathsf{Tr}(F)(x,y)$ then every $\star$-minimal starting move on F is optimal for $\mathsf{Tr}(F)(x,y)$.

They suffice for the conclusion of the theorem, as we can choose T to be the tree where an optimal starting move is played.

Let $\mathsf{G} = \mathsf{Tr}(F)$, $\mathsf{G}_T = \mathsf{Tr}(T)$, and $\mathsf{G}_U = \mathsf{Tr}(U)$. Thus $\mathsf{G} = \mathsf{G}_T + \mathsf{G}_U$. Let g_0 and h_0 be the initial positions in G_T and G_U respectively. If $\hat{w}(g_0) < \hat{w}(h_0)$ then the move taking the root of T is the only $\star$-minimal starting move on F and thus the conclusion holds trivially. Thus assume

$$\hat{w}(g_0) \geqslant \hat{w}(h_0).$$

Let $g_0 \dots g_m$ be a $\star$-minimal line of play in G_T such that $m \geqslant 1$ and

$$\begin{aligned} \hat{w}(g_i) &\leqslant \hat{w}(g_0) \quad \text{for } 0 \leqslant i < m, \\ \hat{w}(g_m) &\geqslant \hat{w}(g_0). \end{aligned} \tag{6.13}$$

Define

$$s_i = \partial(g_i) = \sum_{k=1}^{i} (-1)^k w(u_k),$$

where u_k denotes the vertex of T taken by the move $g_{k-1}g_k$. By the induction hypothesis, $g_0 \dots g_m$ is an optimal line of play for G_T and every $\mathsf{G}_T(x,y)$. Thus we can apply Lemma 6.2 to G_T and $g_0 \dots g_m$. It follows that

$$s_i \leqslant \hat{w}(g_0) \quad \text{for } 1 \leqslant i \leqslant m \text{ and } i \text{ odd}, \tag{6.14}$$

$$s_i \geqslant \hat{w}(g_i) \quad \text{for } 1 \leqslant i < m \text{ and } i \text{ odd}, \tag{6.15}$$

$$s_i \geqslant 0 \quad \text{for } 0 \leqslant i \leqslant m \text{ and } i \text{ even}, \tag{6.16}$$

$$s_m = \hat{w}(g_0) \quad \text{if } m \text{ is odd}, \tag{6.17}$$

$$s_m = 0 \quad \text{if } m \text{ is even}. \tag{6.18}$$

A position g_i with $1 \leqslant i \leqslant m$ is *principal* if $\hat{w}(g_k) \leqslant \hat{w}(g_i)$ for $1 \leqslant k \leqslant i$. In particular, g_1 and g_m are principal positions. For any two consecutive principal positions g_i and g_{i+r}, by the induction hypothesis, $g_i \dots g_{i+r}$ is an optimal line of play for $\mathsf{G}_T(g_i)$ and every $\mathsf{G}_T(g_i)(x,y)$, and thus by Lemma 6.2 we have

$$(-1)^i(s_{i+k} - s_i) \leqslant \hat{w}(g_i) \quad \text{for } 1 \leqslant k \leqslant r \text{ and } k \text{ odd}, \tag{6.19}$$

$$(-1)^i(s_{i+k} - s_i) \geqslant 0 \quad \text{for } 0 \leqslant k \leqslant r \text{ and } k \text{ even}, \tag{6.20}$$

$$(6.21)\qquad (-1)^i(s_{i+r}-s_i)=\hat{w}(g_i) \quad \text{if } r \text{ is odd,}$$

$$(6.22)\qquad (-1)^i(s_{i+r}-s_i)=0 \quad \text{if } r \text{ is even.}$$

Let $h_0 \dots h_n$ be a $\star$-minimal line of play in G_U such that $n \geqslant 1$ and

$$(6.23)\qquad \begin{aligned} \hat{w}(h_j) &\leqslant \hat{w}(g_0) \quad \text{for } 0 \leqslant j < n, \\ \hat{w}(h_n) &\geqslant \hat{w}(g_0). \end{aligned}$$

Define

$$t_j = \partial(h_j) = \sum_{k=1}^{j} (-1)^k w(v_k),$$

where v_k denotes the vertex of U taken by the move $h_{k-1}h_k$. A position h_j with $0 \leqslant j \leqslant n$ is *principal* if $\hat{w}(h_k) \leqslant \hat{w}(h_j)$ for $0 \leqslant k \leqslant j$. In particular, h_0 and h_n are principal positions. For any two consecutive principal positions h_j and h_{j+r}, by the induction hypothesis, $h_j \dots h_{j+r}$ is an optimal line of play for $\mathsf{G}_U(h_j)$ an every $\mathsf{G}_U(h_j)(x,y)$, and thus by Lemma 6.2 we have

$$(6.24)\qquad (-1)^j(t_{j+k}-t_j) \leqslant \hat{w}(h_j) \quad \text{for } 1 \leqslant k \leqslant r \text{ and } k \text{ odd,}$$

$$(6.25)\qquad (-1)^j(t_{j+k}-t_j) \geqslant 0 \quad \text{for } 0 \leqslant k \leqslant r \text{ and } k \text{ even,}$$

$$(6.26)\qquad (-1)^j(t_{j+r}-t_j) = \hat{w}(h_j) \quad \text{if } r \text{ is odd,}$$

$$(6.27)\qquad (-1)^j(t_{j+r}-t_j) = 0 \quad \text{if } r \text{ is even.}$$

Let P be the line of play in G of length $m+n$ obtained by first making all moves of $h_0 \dots h_n$ on U and then making all moves of $g_0 \dots g_m$ on T. We denote by $p_{i,j}$ the position on P whose projections onto G_T and G_U are g_i and h_j respectively. Thus $p_{0,0}$ is the initial position in G, and $P = p_{0,0} \dots p_{0,n} \dots p_{m,n}$. It follows from (6.23) and $\star$-minimality of $g_0 \dots g_m$ and $h_0 \dots h_n$ that P is $\star$-minimal. Clearly

$$(6.28)\qquad \hat{w}(p_{i,j}) = \min\{\hat{w}(g_i), \hat{w}(h_j)\},$$

$$(6.29)\qquad \partial(p_{i,j}) = t_j + (-1)^n s_i.$$

We also construct a line of play Q in G of length $m+n$ by making the moves of $g_0 \dots g_m$ on T and the moves of $h_0 \dots h_n$ on U in the following order. We start with the move g_0g_1 on T. Then, repeatedly, from a position whose projections onto G_T and G_U are g_i and h_j (respectively) we make the move g_ig_{i+1} on T if $\hat{w}(g_i) < \hat{w}(h_j)$, the move h_jh_{j+1} on U if $\hat{w}(g_i) > \hat{w}(h_j)$, or any of the two if $\hat{w}(g_i) = \hat{w}(h_j)$. We denote by $q_{i,j}$ the position on Q whose projections onto G_T and G_U are g_i and h_j respectively. Thus $q_{0,0}$ is the initial position in G, and the position following $q_{0,0}$ is $q_{1,0}$. Clearly, Q is $\star$-minimal except for the first move, the last position on Q is $q_{m,n}$, and

$$(6.30)\qquad \hat{w}(q_{i,j}) = \min\{\hat{w}(g_i), \hat{w}(h_j)\}.$$

We call a position $q_{i,j}$ on Q *principal* if both g_i and h_j are principal. In particular, $q_{1,0}$ and $q_{m,n}$ are principal positions. Every subpath of Q between two consecutive principal positions is of the form $q_{i,j} \dots q_{i+r,j}$ or $q_{i,j} \dots q_{i,j+r}$. It follows from the definitions of principal positions on $g_1 \dots g_m$ and $h_0 \dots h_n$ that if $q_{i,j}$ is principal and q' is a position on Q before $q_{i,j}$ then

$$(6.31)\qquad \hat{w}(q') \leqslant \hat{w}(q_{i,j}).$$

By (6.30), (6.13), and (6.23), if $q_{i,j}$ is principal then

$$(6.32)\qquad \hat{w}(q_{i,j}) \leqslant w(g_0) \quad \text{if } q_{i,j} \neq q_{m,n},$$

$$(6.33)\qquad \hat{w}(q_{m,n}) \geqslant w(g_0).$$

Claim 6.3.1. *For every principal position $q_{i,j}$ on Q we have*

$$\partial(q_{i,j}) \geqslant \begin{cases} s_i + t_j & \text{if } i \text{ or } j \text{ is even,} \\ s_i + t_j - 2\hat{w}(q') & \text{if both } i \text{ and } j \text{ are odd,} \end{cases} \tag{6.34}$$

where q' is the last principal position on Q before $q_{i,j}$.

Proof. We prove Claim 6.3.1 by induction on the sequence of principal positions on Q. We have

$$\partial(q_{1,0}) = -w(u_1) = s_1,$$

so the conclusion holds for $q_{1,0}$. Now, suppose that it holds for $q_{i,j}$, that is, we have (6.34). We show that the conclusion holds also for the next principal position after $q_{i,j}$, which is either $q_{i+r,j}$ or $q_{i,j+r}$ for some r. Suppose that the next principal position is $q_{i+r,j}$. Thus we have

$$\hat{w}(q_{i,j}) = \hat{w}(g_i). \tag{6.35}$$

We also have

$$\partial(q_{i+r,j}) = \partial(q_{i,j}) + (-1)^j(s_{i+r} - s_i). \tag{6.36}$$

Our goal is to show

$$\partial(q_{i+r,j}) \geqslant \begin{cases} s_{i+r} + t_j & \text{if } i+r \text{ or } j \text{ is even,} \\ s_{i+r} + t_j - 2\hat{w}(q_{i,j}) & \text{if both } i+r \text{ and } j \text{ are odd.} \end{cases} \tag{6.37}$$

If j is even then we have

$$\begin{aligned} \partial(q_{i+r,j}) &\geqslant s_i + t_j + (s_{i+r} - s_i) && \text{by (6.36) and (6.34)} \\ &= s_{i+r} + t_j. \end{aligned}$$

Now, suppose j is odd. If i and r are even then

$$\begin{aligned} \partial(q_{i+r,j}) &\geqslant s_i + t_j - (s_{i+r} - s_i) && \text{by (6.36) and (6.34)} \\ &= s_i + t_j + (s_{i+r} - s_i) && \text{by (6.22)} \\ &= s_{i+r} + t_j. \end{aligned}$$

If i is even and r is odd then

$$\begin{aligned} \partial(q_{i+r,j}) &\geqslant s_i + t_j - (s_{i+r} - s_i) && \text{by (6.36) and (6.34)} \\ &= s_{i+r} + t_j - 2(s_{i+r} - s_i) \\ &= s_{i+r} + t_j - 2\hat{w}(q_{i,j}) && \text{by (6.21) and (6.35).} \end{aligned}$$

If i is odd and r is even then

$$\begin{aligned} \partial(q_{i+r,j}) &\geqslant s_i + t_j - 2\hat{w}(q') - (s_{i+r} - s_i) && \text{by (6.36) and (6.34)} \\ &\geqslant s_i + t_j - 2\hat{w}(q_{i,j}) + (s_{i+r} - s_i) && \text{by (6.31) and (6.22)} \\ &= s_{i+r} + t_j - 2\hat{w}(q_{i,j}), \end{aligned}$$

where q' is the last principal position before $q_{i,j}$. Finally, if i and r are odd then

$$\begin{aligned} \partial(q_{i+r,j}) &\geqslant s_i + t_j - 2\hat{w}(q') - (s_{i+r} - s_i) && \text{by (6.36) and (6.34)} \\ &= s_{i+r} + t_j - 2\hat{w}(q') - 2(s_{i+r} - s_i) \\ &= s_{i+r} + t_j - 2\hat{w}(q') + 2\hat{w}(q_{i,j}) && \text{by (6.21) and (6.35)} \\ &\geqslant s_{i+r} + t_j && \text{by (6.31),} \end{aligned}$$

where q' is the last principal position on Q before $q_{i,j}$. We have thus proved (6.37) for the case that the next principal position after $q_{i,j}$ is $q_{i+r,j}$. By an analogous argument but using (6.26) and (6.27) instead of (6.21) and (6.22) we prove

$$\partial(q_{i,j+r}) \geqslant \begin{cases} s_i + t_{j+r} & \text{if } i \text{ or } j+r \text{ is even,} \\ s_i + t_{j+r} - 2\hat{w}(q_{i,j}) & \text{if both } i \text{ and } j+r \text{ are odd} \end{cases}$$

for the case that the next principal position after $q_{i,j}$ is $q_{i,j+r}$. □

Claim 6.3.2. $\partial(p_{m,n}) \leqslant \partial(q_{m,n})$.

Proof. By Claim 6.3.1 and (6.32) we have

$$\partial(q_{m,n}) \geqslant \begin{cases} s_m + t_n & \text{if } m \text{ or } n \text{ is even,} \\ s_m + t_n - 2\hat{w}(g_0) & \text{if both } m \text{ and } n \text{ are odd.} \end{cases}$$

If n is even then by (6.29) we have

$$\partial(p_{m,n}) = t_n + s_m \leqslant \partial(q_{m,n}).$$

If n is odd and m is even then by (6.29) and (6.18) we have

$$\partial(p_{m,n}) = t_n - s_m = t_n + s_m \leqslant \partial(q_{m,n}).$$

If both n and m are odd then by (6.29) and (6.17) we have

$$\partial(p_{m,n}) = t_n - s_m = t_n + s_m - 2\hat{w}(g_0) \leqslant \partial(q_{m,n}).$$ □

Suppose that the starting move $q_{0,0}q_{1,0}$ taking the root of T is optimal for G. Since $q_{1,0} \dots q_{m,n}$ is a $\star$-minimal line of play in $\mathsf{G}(q_{1,0})$, by the induction hypothesis it is optimal for $\mathsf{G}(q_{1,0})$. Thus Q is an optimal line of play in G. We want to prove that a $\star$-minimal starting move in G other than $q_{0,0}q_{1,0}$ is optimal for G as well. The considered starting move is made on U and thus we can assume that it is the move $p_{0,0}p_{0,1}$, as the first move on $h_0 \dots h_n$ is chosen arbitrarily with the only restriction that it is $\star$-minimal. Since $p_{0,1} \dots p_{m,n}$ is a $\star$-minimal line of play in $\mathsf{G}(p_{0,1})$, by the induction hypothesis it is optimal for $\mathsf{G}(p_{0,1})$. Thus P consists of the first move $p_{0,0}p_{0,1}$ followed by optimal moves. The lines of play P and Q lead to positions $p_{m,n}$ and $q_{m,n}$ respectively, the subgames at which are isomorphic. This and Claim 6.3.2 show that the starting move $p_{0,0}p_{0,1}$ is also optimal for G.

Now, suppose that for some x, y with $y \leqslant 0$ the starting move $q_{0,0}q_{1,0}$ is optimal for $\mathsf{G}(x, y)$. It again follows from induction hypothesis applied to $\mathsf{G}(q_{1,0})$ that Q is optimal for $\mathsf{G}(x, y)$. We want to prove that a $\star$-minimal starting move in G other than $q_{0,0}q_{1,0}$ is optimal for $\mathsf{G}(x, y)$ as well. As before, we assume that the considered starting move is $p_{0,0}p_{0,1}$ and show that P is optimal for $\mathsf{G}(x, y)$. Suppose the contrary, that P is not optimal for $\mathsf{G}(x, y)$. Thus Q leads to Alice's win, while P leads to Bob's win. We show that every move on P as well as on Q is valid for $\mathsf{G}(x, y)$. It follows that $\mathsf{G}(x, y)$ played along P or Q reaches the position $p_{m,n}$ or $q_{m,n}$ respectively. This and Claim 6.3.2 contradict the assumption that Q leads to Alice's win and P leads to Bob's win.

Claim 6.3.3. *All principal positions $p_{0,j}$ and $q_{i,j}$ are valid for $\mathsf{G}(x,y)$. In particular, we have*

$$\partial(p_{0,j}) \leqslant x \quad \text{if } p_{0,j} \text{ is odd,} \tag{6.38}$$
$$\partial(p_{0,j}) \geqslant y \quad \text{if } p_{0,j} \text{ is even,} \tag{6.39}$$
$$\partial(p_{0,j}) \geqslant y + \hat{w}(p_{0,j}) \quad \text{if } p_{0,j} \text{ is odd,} \tag{6.40}$$
$$\partial(q_{i,j}) \leqslant x \quad \text{if } q_{i,j} \text{ is odd,} \tag{6.41}$$
$$\partial(q_{i,j}) \geqslant y \quad \text{if } q_{i,j} \text{ is even,} \tag{6.42}$$
$$\partial(q_{i,j}) \leqslant x - \hat{w}(q_{i,j}) \quad \text{if } q_{i,j} \text{ is even.} \tag{6.43}$$

Proof. First we show how (6.38)–(6.43) follow from validity of $p_{0,j}$ and $q_{i,j}$. Suppose that $p_{0,j}$ is valid for $\mathsf{G}(x,y)$. This clearly implies (6.38) and (6.39). To see (6.40), suppose $p_{0,j}$ is odd. Since the game $\mathsf{G}(x,y)$ played along P is won by Bob and does not end before $p_{0,j}$, the first player wins the game $\mathsf{G}(p_{0,j})(\partial(p_{0,j}) - y, \partial(p_{0,j}) - x)$ and thus $\mathsf{G}(p_{0,j})(\partial(p_{0,j}) - y, 0)$ as well. This and (6.2) yield

$$\partial(p_{0,j}) - y \geqslant \mathrm{val}^\star \mathsf{G}(p_{0,j}) \geqslant \hat{w}(p_{0,j}).$$

Now, suppose that $q_{i,j}$ is valid for $\mathsf{G}(x,y)$. Again, (6.41) and (6.42) clearly follow. To see (6.43), suppose $q_{i,j}$ is even. Since the game $\mathsf{G}(x,y)$ played along Q is won by Alice and does not end before $q_{i,j}$, the first player wins $\mathsf{G}(q_{i,j})(x - \partial(q_{i,j}), y - \partial(q_{i,j}))$ and thus $\mathsf{G}(q_{i,j})(x - \partial(q_{i,j}), 0)$ as well. This and (6.2) yield

$$x - \partial(q_{i,j}) \geqslant \mathrm{val}^\star \mathsf{G}(q_{i,j}) \geqslant \hat{w}(q_{i,j}).$$

It remains to prove that $p_{0,j}$ and $q_{i,j}$ are valid for $\mathsf{G}(x,y)$. Since we have assumed that $\mathsf{G}(x,y)$ played along Q is won by Alice and played along P is won by Bob, no even position on Q or odd position on P can be final in $\mathsf{G}(x,y)$. Thus it suffices to prove that even positions on Q and odd positions on P are valid for $\mathsf{G}(x,y)$.

Like in the proof of Claim 6.3.1, we argue by induction on the sequence of principal positions on Q. The conclusion holds trivially for $q_{1,0}$. Suppose that it holds for a principal position $q_{i,j} \neq q_{m,n}$. We show that it also holds for the next principal position, which is either $q_{i+r,j}$ or $q_{i,j+r}$ for some r.

First we show that every even position q on Q between $q_{i,j}$ and the next principal position after $q_{i,j}$ satisfies $\partial(q) \geqslant y$. Suppose $q_{i,j}$ is even. We have

$$\partial(q) = \begin{cases} \partial(q_{i,j}) + (-1)^i (s_{i+k} - s_i) & \text{if } q = q_{i+k,j}, \\ \partial(q_{i,j}) + (-1)^j (t_{j+k} - t_j) & \text{if } q = q_{i,j+k}, \end{cases}$$

where k is even. It follows that

$$\begin{aligned} \partial(q) &\geqslant \partial(q_{i,j}) && \text{by (6.20) and (6.25)} \\ &\geqslant y && \text{by (6.42).} \end{aligned}$$

Now, suppose $q_{i,j}$ is odd. We have

$$\partial(q) = \begin{cases} \partial(q_{i,j}) - (-1)^i (s_{i+k} - s_i) & \text{if } q = q_{i+k,j}, \\ \partial(q_{i,j}) - (-1)^j (t_{j+k} - t_j) & \text{if } q = q_{i,j+k}, \end{cases}$$

where k is odd. It follows that

$$\begin{aligned} \partial(q) &\geqslant \partial(q_{i,j}) - \hat{w}(q_{i,j}) && \text{by (6.19) and (6.24)} \\ &\geqslant s_i + t_j - \hat{w}(q_{i,j}) && \text{by Claim 6.3.1} \\ &= \partial(p_{0,j}) + s_i - \hat{w}(q_{i,j}) && \text{by (6.29).} \end{aligned}$$

If i is odd and j is even then we consequently have

$$\begin{aligned}\partial(q) &\geqslant \partial(p_{0,j}) + s_i - \min\{\hat{w}(g_i), \hat{w}(g_0)\} && \text{by (6.30) and (6.32)}\\ &\geqslant \partial(p_{0,j}) && \text{by (6.15) and (6.17)}\\ &\geqslant y && \text{by (6.39).}\end{aligned}$$

If i is even and j is odd then we consequently have

$$\begin{aligned}\partial(q) &\geqslant \partial(p_{0,j}) - \hat{w}(q_{i,j}) && \text{by (6.16)}\\ &\geqslant \partial(p_{0,j}) - \min\{\hat{w}(h_j), \hat{w}(g_0)\} && \text{by (6.30) and (6.32)}\\ &= \partial(p_{0,j}) - \hat{w}(p_{0,j}) && \text{by (6.28)}\\ &\geqslant y && \text{by (6.40).}\end{aligned}$$

Now we suppose that the next principal position after $q_{i,j}$ is $q_{i,j+r}$, which in particular implies

$$\hat{w}(q_{i,j}) = \hat{w}(h_j), \tag{6.44}$$

and show that every odd position $p_{0,j+k}$ with $1 \leqslant k \leqslant r$ satisfies $\partial(p_{0,j+k}) \leqslant x$, which by (6.29) is equivalent to $t_{j+k} \leqslant x$. Suppose j is odd. It follows that k is even. We have

$$\begin{aligned}t_{j+k} &\leqslant t_j && \text{by (6.25)}\\ &= \partial(p_{0,j}) && \text{by (6.29)}\\ &\leqslant x && \text{by (6.38).}\end{aligned}$$

Now, suppose j is even. It follows that k is odd. We have

$$\begin{aligned}t_{j+k} &\leqslant t_j + \hat{w}(h_j) && \text{by (6.24)}\\ &= t_j + \hat{w}(q_{i,j}) && \text{by (6.44).}\end{aligned}$$

If i is odd then we consequently have

$$\begin{aligned}t_{j+k} &\leqslant t_j + \min\{\hat{w}(g_i), \hat{w}(g_0)\} && \text{by (6.30) and (6.32)}\\ &\leqslant t_j + s_i && \text{by (6.15) and (6.17)}\\ &\leqslant \partial(q_{i,j}) && \text{by Claim 6.3.1}\\ &\leqslant x && \text{by (6.41).}\end{aligned}$$

If i is even then we consequently have

$$\begin{aligned}t_{j+k} &\leqslant s_i + t_j + \hat{w}(q_{i,j}) && \text{by (6.16)}\\ &\leqslant \partial(q_{i,j}) + \hat{w}(q_{i,j}) && \text{by Claim 6.3.1}\\ &\leqslant x && \text{by (6.43).}\end{aligned}$$

□

It follows from Claim 6.3.3 that the positions on Q and the positions on P up to $p_{0,n}$ are valid for $\mathsf{G}(x, y)$. To finish the whole proof we still need to argue that every position on P after $p_{0,n}$ is valid for $\mathsf{G}(x, y)$. As before, it suffices to prove $\partial(p_{i,n}) \leqslant x$ for every odd position $p_{i,n}$ with $1 \leqslant i \leqslant m$. Suppose that i is even and n is odd. We have

$$\begin{aligned}\partial(p_{i,n}) &= t_n - s_i && \text{by (6.29)}\\ &\leqslant t_n && \text{by (6.16)}\\ &= \partial(p_{0,n}) && \text{by (6.29)}\\ &\leqslant x && \text{by (6.38).}\end{aligned}$$

Now, suppose that i is odd and n is even. We have

$$\begin{aligned} \partial(p_{i,n}) &= t_n + s_i && \text{by (6.29)} \\ &\leqslant t_n + \hat{w}(g_0) && \text{by (6.14).} \end{aligned}$$

If m is odd then it follows that

$$\begin{aligned} \partial(p_{i,n}) &\leqslant t_n + s_m && \text{by (6.17)} \\ &\leqslant \partial(q_{m,n}) && \text{by Claim 6.3.1} \\ &\leqslant x && \text{by (6.41).} \end{aligned}$$

If m is even then we consequently have

$$\begin{aligned} \partial(p_{i,n}) &\leqslant t_n + s_m + \hat{w}(q_{m,n}) && \text{by (6.18) and (6.33)} \\ &\leqslant \partial(q_{m,n}) + \hat{w}(q_{m,n}) && \text{by Claim 6.3.1} \\ &\leqslant x && \text{by (6.43).} \end{aligned}$$

This completes the proof of the theorem. □

6.3. Algorithm computing optimal strategies

We are ready to provide a polynomial-time algorithm computing optimal strategies of both players for game T on a weighted tree. What we expect of such an algorithm is not to compute the entire strategy tree, which is usually of exponential size, but to find an optimal move at any given position in the game. It is enough to compute an optimal first move in the game $\mathsf{T}(T)$ on a weighted tree T or the game $\mathsf{Tr}(F)$ on a weighted rooted forest F, as the subgame of $\mathsf{T}(T)$ at any position other than the initial one is of the form $\mathsf{Tr}(F)$ for a rooted subforest F of T.

Let F be a weighted rooted forest oriented naturally so that every vertex is reachable by an oriented path from a root. Clearly, for every position p in $\mathsf{Tr}(F)$ the components of the subforest of F induced on the untaken vertices at p are of the form $F(u)$ for $u \in V(F)$. By Theorem 6.3 an optimal strategy in $\mathsf{Tr}(F)$ is to always take an available vertex u with minimum $\mathrm{val}^\star \mathsf{Tr}(F(u))$. Thus what our algorithm actually needs to compute are the values $\mathrm{val}^\star \mathsf{Tr}(F(u))$ for all $u \in V(F)$. We compute these values bottom-up, starting from leaves and processing a vertex u after $\mathrm{val}^\star \mathsf{Tr}(F(v))$ is known for all $v \in V(F(u)) - \{u\}$. Together with each value $\mathrm{val}^\star \mathsf{Tr}(F(u))$ we compute the order $u^1, \ldots, u^k$ in which the vertices of $F(u)$ are taken by some $\star$-minimal line of play in $\mathsf{Tr}(F(u))$.

To process a vertex u, we compute $u^1, \ldots, u^k$ first. Let $v_1, \ldots, v_d$ be the children of u in F. For $1 \leqslant i \leqslant d$, let $v_i^1, \ldots, v_i^{k_i}$ denote the order of the vertices of $F(v_i)$ computed when processing v_i. The sequence $u^1, \ldots, u^k$ arises by putting u first and then merging the sequences $v_i^1, \ldots, v_i^{k_i}$, as follows:

$u^1 := u$
$j_1, \ldots, j_d := 1$
for $j := 2$ **to** k **do**
  choose $i \in \{1, \ldots, d\}$ so that $j_i \leqslant k_i$ and $\mathrm{val}^\star \mathsf{Tr}(F(v_i^{j_i}))$ is minimal
  $u^j := v_i^{j_i}$
  $j_i := j_i + 1$
end

By Theorem 6.3 the line of play in $\mathsf{Tr}(F(u))$ taking the vertices $u^1, \dots, u^k$ in this order is optimal for every $\mathsf{Tr}(F(u))(x,y)$. Therefore, to evaluate $\mathrm{val}^\star\,\mathsf{Tr}(F(u))$, we can use the following formula, which follows easily from the definition of $\mathrm{val}^\star$:

$$\mathrm{val}^\star\,\mathsf{Tr}(F(u)) = \begin{cases} \max\bigl\{\partial(p_j) : 1 \leqslant j \leqslant \ell + 1 \text{ and } j \text{ is odd}\bigr\} & \text{if } \ell < k, \\ \infty & \text{if } \ell = k, \end{cases}$$

where ℓ is the greatest even index with $0 \leqslant \ell \leqslant k$ and $\partial(p_\ell) \geqslant 0$, and $\partial(p_j)$ is computed as

$$\partial(p_j) = \sum_{i=1}^{j} (-1)^i w(u^i).$$

This gives us the following procedure to compute $\mathrm{val}^\star\,\mathsf{Tr}(F(u))$:

$partial := 0$
$result := -\infty$
for $j := 1$ **to** k **do**
  $partial := partial + (-1)^j w(u^j)$
  if j is odd **then** $result := \max\{result, partial\}$
  if j is even **and** $partial < 0$ **then return** $result$
end
if k is odd **then return** $result$
if k is even **then return** ∞

Computing the sequence $u^1, \dots, u^k$ according to the first procedure above takes time $O(dk+1)$, where d is the number of children of u. Computing $\mathrm{val}^\star\,\mathsf{Tr}(F(u))$ according to the second procedure takes time $O(k)$. Thus the total time required for all computation amounts to $O(n^2)$, where $n = |V(F)|$, as the total number of children of all vertices of F is $n - |\mathrm{Comp}\,F|$.

When we have computed optimal lines of play in $\mathrm{val}^\star\,\mathsf{Tr}(T)$ for all components T of F, we can merge them like in the first procedure above to obtain an optimal line of play in $\mathrm{val}^\star\,\mathsf{Tr}(F)$. The players' outcomes at the final position of this line of play are the players' values of $\mathsf{Tr}(F)$.

Theorem 6.4. *There is an algorithm that given a weighted rooted forest F computes both players' values of $\mathsf{Tr}(F)$ and an optimal move at the initial position in $\mathsf{Tr}(F)$ in time $O(n^2)$.* □

Now, let T be a weighted tree, and consider the game $\mathsf{T}(T)$. When Alice decides to take v in her first move, the game becomes identical to $\mathsf{Tr}(T)$ with T considered as rooted at v. To compute the players' values of $\mathsf{T}(T)$ or an optimal first move in $\mathsf{T}(T)$, we just check all $|V(T)|$ possible first moves, compute the players' outcomes using the algorithm for game Tr, and choose a move that maximizes the outcome of Alice.

Theorem 6.5. *There is an algorithm that given a weighted tree T computes both players' values of $\mathsf{T}(T)$ and an optimal move at the initial position in $\mathsf{T}(T)$ in time $O(n^3)$.* □

Chapter 7

Game R on even trees

In this short chapter we show that Alice can secure at least $\frac{1}{4}$ of the total weight of any tree with an even number of vertices in game R. This result is a joint work with Piotr Micek and is contained in [13].

Theorem 7.1. *For every weighted tree T with an even number of vertices Alice has a strategy in $\mathsf{R}(T)$ to collect vertices of total weight at least $\frac{1}{4}w(T)$.*

Proof. A *two-colored tree* is a tree whose vertices are colored black or white so that no two adjacent vertices have the same color. We prove that for every two-colored weighted tree T with an even number of vertices Alice can secure at least $\frac{1}{2}$ of the total weight of black vertices. This suffices for the conclusion of the theorem, as we can choose the colors so that the total weight of black vertices is at least the total weight of white ones.

The proof goes by induction on $|V(T)|$. For a tree with two vertices the statement is trivial. Thus suppose that T has at least four vertices. We construct a strategy for Alice to reach an even position at which her partial outcome is at least Bob's. We distinguish two cases: either T has a black leaf or it does not.

If T has a black leaf then Alice takes a heaviest one. With this move she does not uncover any new black vertex, so Bob can only respond with a white vertex or a black vertex with smaller or equal weight. In both scenarios Alice gains at least as much as Bob from the black part, and for the rest of the tree the induction hypothesis is applied.

Thus suppose that all leaves of T are white. It follows that there is a black vertex of degree greater than 2: if all black vertices have degree 2 then the total number of edges is even, so T has an odd number of vertices, which contradicts the assumption. Let K be the minimal subtree of T containing all black vertices of degree greater than 2. Thus all leaves of K are black and have degree greater than 2. We call K the *core* of T, and we call components of $T - V(K)$ simply *components*. The *root* of a component is its only vertex adjacent to K. A similar argument with counting edges shows that every component with white root has an odd number of vertices and every component with black root has an even number of vertices. Since all leaves of the core are black and have degree greater than 2, at least two of their neighbors must be roots of odd components. Therefore, taking a vertex from the core is possible only after at least two odd components have been entirely taken.

Let $C_1, \dots, C_k$ be the components, and assume that C_1 is an odd component with the least total weight of black vertices among all odd components. Alice starts with any vertex from C_1. We keep the following invariants for odd positions until the induction hypothesis is applied:

(1) An odd number of vertices have been taken from C_1.
(2) For $i \neq 1$, an even number of vertices have been taken from C_i.

(3) For $i \neq 1$, all vertices taken from C_i have been taken by Alice.
(4) For $i \neq 1$, all available vertices in C_i are white.

Note that by (2) no odd component other than C_1 has been entirely taken, and thus no vertex from the core is available. Therefore, Bob can choose only from the vertices in the components.

Suppose that Bob takes a vertex from C_1. Alice answers by taking another vertex of C_1. Such a vertex exists by (1) and the assumption that C_1 has an odd number of vertices. All invariants are clearly preserved.

Now, suppose that Bob takes a vertex v from C_i with $i \neq 1$. By (4) v is white. Unless v is the root of C_i, Bob's move uncovers exactly one black vertex in C_i, which becomes a new leaf. Alice takes this black vertex. All vertices that remain available in C_i are white, exactly as (4) states. If v is the root of C_i then C_i has an odd number of vertices and all of them have been taken. By (3) all black vertices taken by Bob are in C_1, and Alice has taken all black vertices from C_i. Since the weight of black vertices in C_i is at least the weight of black vertices in C_1, Alice has collected at least as much as Bob from the black part. For the remaining tree the induction hypothesis is applied. □

As already mentioned in the introduction (see Theorem 1.6), the result above has been improved by Seacrest and Seacrest [16], who proved that for every even tree T Alice has a strategy in $\mathsf{R}(T)$ to gather at least $\frac{1}{2}w(T)$. However, their proof of this result is non-constructive and does not yield an efficient algorithm realizing the claimed strategy. Our proof above can be transformed into a polynomial-time algorithm realizing a strategy that allows Alice to take $\frac{1}{4}w(T)$ in $\mathsf{R}(T)$ for any tree T with an even number of vertices. A constructive strategy allowing Alice to take $\frac{1}{2}$ of the total weight of any subdivided star (i.e. tree with at most one vertex of degree greater than two) with an even number of vertices is presented in [13].

Chapter 8

Structural properties of weighted graphs

For a weighted graph G we define

$$\operatorname{balance} G = w(G) - \max_{C \in \operatorname{Comp} G} w(C).$$

Our goal in this chapter is to prove that for a suitable constant $c_n > 0$ every weighted connected graph G with no subdivision of K_n contains one of the following structures:

- a connected set S of vertices such that $\operatorname{balance}(G - S) \geqslant c_n \cdot w(G)$;
- a set S of vertices with $w(S) \geqslant c_n \cdot w(G)$ such that $G\{S\}$ is a cycle.

They provide a base for strategies of Alice developed in the next chapter. First, we show that every weighted connected graph G contains one of the structures above or a connected set S of vertices with $w(N_G(S)) \geqslant c_n \cdot w(G)$. Then, we reduce the latter case to the first two for graphs with forbidden subdivision of K_n.

8.1. General graphs

For an oriented path P let $<_P$ denote the linear ordering of the vertices along P. For vertices u and v of an oriented path P we define

$$(u, v)_P = \big\{x \in V(P) : u <_P x <_P v\big\},$$
$$[u, v]_P = \big\{x \in V(P) : u \leqslant_P x \leqslant_P v\big\}.$$

An oriented path P in an oriented graph G is *Hamiltonian* if $V(P) = V(G)$. A vertex x of an oriented graph G with a Hamiltonian path P is *P-covered* by an edge $uv \in E(G)$ if $x \in (u, v)_P$.

Lemma 8.1. *Let G be an acyclic oriented graph with at least two vertices and with a Hamiltonian path P beginning at s and ending at t. If all vertices from $V(G) - \{s, t\}$ are P-covered then there are two s,t-paths Q_0 and Q_1 in G such that $V(Q_0) \cap V(Q_1) = \{s, t\}$.*

Proof. We construct inductively two sequences $u_0, \ldots, u_k$ and $v_0, \ldots, v_k$ of vertices of G such that $u_0 = v_0 = s$ and $u_k = v_k = t$, maintaining the following invariants:

(1) $u_i v_{i+1} \in E(G)$,
(2) $v_{i-1} \leqslant_P u_i <_P v_i <_P v_{i+1}$ for $i \geqslant 1$,
(3) no edge $xy \in E(G)$ satisfies $v_i, v_{i+1} \in (x, y)_P$.

Set $u_0 = v_0 = s$ and choose v_1 to be greatest in $<_P$ such that $u_0 v_1 \in E(G)$. The invariants are clearly satisfied for $i = 0$. Then, for $j \geqslant 1$ repeat the following step. If $v_j = t$ then set $u_j = t$ and $k = j$, and terminate the construction. Otherwise choose two vertices u_i and v_{i+1} so that (1) and (2) are satisfied and additionally v_{j+1} is greatest possible in $<_P$. Such an edge exists, because v_j is P-covered by an edge, which cannot start before v_{j-1} by the invariant (3) with $i = j - 1$. Now, to see that (3) holds for $i = j$, note that any edge $xy \in E(G)$ such that $v_j, v_{j+1} \in (x, y)_P$ would contradict maximality of v_{j+1} in the choice of u_j and v_{j+1}.

It follows from (2) that the intervals $[v_{i-1}, u_i]_P$ are non-empty and pairwise disjoint. Let Q_0 consist of the subpaths of P induced by the intervals $[v_{i-1}, u_i]_P$ for i even, and of the edges $u_i v_{i+1} \in E(G)$ for i even. Let Q_1 consist of the subpaths of P induced by the intervals $[v_{i-1}, u_i]_P$ for i odd, and of the edges $u_i v_{i+1} \in E(G)$ for i odd. It follows that Q_0 and Q_1 are vertex disjoint except at s and t. □

For vertices u and v of an oriented cycle H we define $(u, v)_H$ to be

- the set of internal vertices of the subpath of H going from u to v if $u \neq v$,
- the set $V(H) - \{u\}$ if $u = v$,

and we also define

$$[u, v)_H = \{u\} \cup (u, v)_H,$$
$$(u, v]_H = (u, v)_H \cup \{v\}.$$

An (oriented) cycle H in an (oriented) graph G is *Hamiltonian* if $V(H) = V(G)$. A vertex x of an oriented graph G with a Hamiltonian cycle H is *H-covered* by an edge $uv \in E(G)$ if $x \in (u, v)_H$.

Recall that according to our definition a single-vertex graph and a graph consisting of two vertices joined by an edge are unoriented cycles.

Lemma 8.2. *There is $c \in (0, 1]$ such that for every weighted graph G containing a Hamiltonian cycle H one of the following holds:*

(1) There is a connected set $S \subseteq V(G)$ such that

$$\text{balance}(H - S) \geqslant c \cdot w(G).$$

(2) There is a set $S \subseteq V(G)$ such that $G\{S\}$ is a cycle and

$$w(S) \geqslant c \cdot w(G).$$

Proof. Set

$$c = 1/5. \tag{8.1}$$

To see that this choice of c satisfies the conclusion, let G be a weighted graph containing a Hamiltonian cycle H. If G has no more than 3 vertices then (2) holds for $S = V(G)$. If some edge $uv \in E(G)$ satisfies balance$(H - \{u, v\}) \geqslant c \cdot w(G)$ then (1) holds for $S = \{u, v\}$. Thus for the remainder of the proof assume that G has at least 4 vertices and every edge $uv \in E(G)$ satisfies

$$\text{balance}(H - \{u, v\}) < c \cdot w(G). \tag{8.2}$$

Suppose that we find two connected sets $S_0, S_1 \subseteq V(G)$ such that

$$w(S_0 \cap S_1) < c \cdot w(G), \tag{8.3}$$
$$\forall C \in \text{Comp}(H - S_k) : w(C) < c \cdot w(G) \quad \text{for } k \in \{0, 1\}. \tag{8.4}$$

By (8.3) we have

$$w(S_0) + w(S_1) = w(S_0 \cup S_1) + w(S_0 \cap S_1) < (1 + c)w(G).$$

If we choose $k \in \{0, 1\}$ so that $w(S_k) \leqslant w(S_{1-k})$, the above yields

$$w(S_k) < \tfrac{1}{2}(1 + c)w(G). \tag{8.5}$$

To conclude, we have

$$\begin{aligned} \text{balance}(H - S_k) &= w(H - S_k) - \max_{C \in \text{Comp}(H - S_k)} w(C) \\ &> \tfrac{1}{2}(1 - c)w(G) - c \cdot w(G) && \text{by (8.5) and (8.4)} \\ &= c \cdot w(G) && \text{by (8.1).} \end{aligned}$$

This shows that (1) holds for $S = S_k$.

We show how to find two connected sets $S_0, S_1 \subseteq V(G)$ satisfying (8.3) and (8.4) or a set $S \subseteq V(G)$ satisfying the conclusion (2) of the lemma. This is enough to complete the proof.

We orient the edges of G as follows. First, we orient the cycle H in any of the two directions. Then, we assign every edge from $E(G) - E(H)$ an orientation uv so that

$$w((u, v)_H) \leqslant w((v, u)_H)$$

(we can choose any orientation if both sides are equal). It follows that every oriented edge $uv \in E(G) - E(H)$ satisfies

$$w((u, v)_H) = \text{balance}(H - \{u, v\}) < c \cdot w(G), \tag{8.6}$$

where the inequality follows from (8.2). From now on, we consider G as an oriented graph and H as an oriented Hamiltonian cycle in G.

Let U be the set of vertices of G that are not H-covered. Suppose $U \neq \emptyset$. Let $u_0, \ldots, u_{n-1}$ be the vertices in U in their order along H and $u_n = u_0$. Every edge $xy \in E(G)$ satisfies $x \in [u_i, u_{i+1})_H$ and $y \in (u_i, u_{i+1}]_H$, as otherwise it would cover a vertex from U. It follows that $G\{U\}$ is the cycle consisting of $u_0, \ldots, u_{n-1}$ in this order. If $w(U) \geqslant c \cdot w(G)$ then (2) holds for $S = U$. Thus assume

$$w(U) < c \cdot w(G). \tag{8.7}$$

For every u_i create a new vertex u'_i and alter all edges of G going into u_i by sending them to u'_i. Thus a new oriented graph G' is obtained. It splits into k pairwise disjoint acyclic oriented graphs $G'_0, \ldots, G'_{k-1}$, each G'_i containing a Hamiltonian path from u_i to u'_{i+1} going through all vertices in $(u_i, u_{i+1})_H$. Every vertex $x \in V(G'_i) - \{u_i, u'_{i+1}\}$ is H-covered by an edge corresponding to the one originally H-covering x in G. Therefore, by Lemma 8.1 each G'_i has two u_i, u'_{i+1}-paths Q^0_i, Q^1_i such that $V(Q^0_i) \cap V(Q^1_i) = \{u_i, u'_{i+1}\}$. For $k \in \{0, 1\}$ let C_k be the cycle in G obtained by taking the union of all Q^k_i and gluing each pair u_i, u'_i back into one vertex. Let $S_k = V(C_k)$. It follows that $S_0 \cap S_1 = U$, and hence by (8.7) we have (8.3). Moreover, every component of $H - S_k$ is entirely contained in $(u, v)_H$ for some edge $uv \in E(C_k)$, and hence (8.4) follows from (8.6). The sets S_0, S_1, being the vertex sets of cycles in G, are clearly connected in G.

Now, suppose $U = \emptyset$. Choose any vertex $v \in V(G)$. Alter all edges H-covering v sending them to v. Thus a new oriented graph $G^\star$ with Hamiltonian cycle H is obtained. Note that if uv is an altered edge then it still satisfies (8.6). The vertex v is not H-covered in $G^\star$. Moreover, all vertices not H-covered in $G^\star$ are H-covered in G by a common edge. Thus by (8.6) the set $U^\star$ of vertices that are not H-covered in $G^\star$ satisfies

$$w(U^\star) < c \cdot w(G).$$

We apply the same argument as for the case $U \neq \emptyset$, but with $G^\star$ and $U^\star$ in place of G and U. By the above it gives us two cycles C_0 and C_1 in $G^\star$ with vertex sets S_0 and S_1, respectively, which satisfy (8.3) and (8.4). Moreover, each C_k can contain only one edge from $E(G^\star) - E(G)$, namely, the one entering v. This shows that each S_k is connected in G. □

Lemma 8.3. *There is $c \in (0, 1]$ such that for every weighted graph G containing a Hamiltonian cycle H and every connected set $A \subseteq V(G)$ one of the following holds:*

(1) There is a connected set $S \subseteq V(G)$ such that $A \subseteq S$ and

$$\text{balance}(G - S) \geqslant c \cdot \text{balance}(H - A).$$

(2) There is a connected set $S \subseteq V(G)$ such that $A \subseteq S$ and

$$w(N_G(S)) \geqslant c \cdot \text{balance}(H - A).$$

Proof. Set

$$c = 1/5. \tag{8.8}$$

To see that this choice of c satisfies the conclusion, let G be a weighted graph containing a Hamiltonian cycle H, and let A be a connected subset of $V(G)$. If $A = V(G)$ or $H - A$ is connected then balance$(H - A) = 0$ and the conclusion holds trivially. Thus assume that $H - A$ has at least two components. Orient H in any of the two directions. Let $<$ be a linear ordering of the vertices of G resulting from breaking the chosen orientation at any edge of the cycle going out of a vertex in A. Thus the greatest vertex in $<$ belongs to A. We partition the set $V(G) - A$ into *blocks* $B_1, \dots, B_k$, which are intervals in the order $<$. We construct them one by one in the order of their indices, as follows. We choose B_i to be the interval $[f_i, u_i)$ such that

- f_i is least in $<$ such that $f_i \notin A \cup B_1 \cup \ldots \cup B_{i-1}$,
- u_i is least in $<$ such that $f_i < u_i$ and $u_i \in N_G[A \cup B_1 \cup \ldots \cup B_{i-1}]$.

Note that if $u_i \notin A$ then $u_i = f_{i+1}$. Now, we partition the family $\{B_1, \dots, B_k\}$ of all blocks into two subfamilies $\mathcal{B}_0$ and $\mathcal{B}_1$ as follows. We process the blocks in the order of their indices. We put B_i into $\mathcal{B}_0$ if $u_i \in N_G[A]$ or u_i is adjacent to at least one of $B_1, \dots, B_{i-1}$ that is already put into $\mathcal{B}_1$. Otherwise we put B_i into $\mathcal{B}_1$. It follows from the presented construction that

- $f_i \in N_G(A)$ or f_i is adjacent to at least one block from $\{B_1, \dots, B_{i-1}\} \cap \mathcal{B}_0$ and at least one block from $\{B_1, \dots, B_{i-1}\} \cap \mathcal{B}_1$;
- no vertex in $B_i - \{f_i\}$ is adjacent to $A \cup B_1 \cup \ldots \cup B_{i-1}$.

For $k \in \{0, 1\}$ define

$$\begin{aligned} A'_k &= A \cup (\bigcup \mathcal{B}_{1-k}), \\ F_k &= \{f_i : B_i \in \mathcal{B}_k\}, \\ A''_k &= A'_k \cup F_k. \end{aligned}$$

Note that $G[A'_k]$ is connected: every block $B_i \in \mathcal{B}_{1-k}$ is connected to A directly or through blocks from $\mathcal{B}_{1-k}$ with indices less than i. Note also that $F_k \subseteq N_G(A'_k)$. If

$$w(F_k) \geqslant c \cdot \text{balance}(H - A)$$

then

$$w(N_G(A'_k)) \geqslant c \cdot \text{balance}(H - A)$$

and hence (2) holds for $S = A'_k$. Thus assume

$$w(F_k) < c \cdot \text{balance}(H - A) \quad \text{for } k \in \{0, 1\}. \tag{8.9}$$

Since $F_k \subseteq N_G(A'_k)$ and $G[A'_k]$ is connected, $G[A''_k]$ is connected too. If

$$\text{balance}(G - A''_k) \geqslant c \cdot \text{balance}(H - A)$$

then (1) holds for $S = A''_k$. Thus assume

$$\text{balance}(G - A''_k) < c \cdot \text{balance}(H - A) \quad \text{for } k \in \{0, 1\}. \tag{8.10}$$

The components of $G - A''_k$ are precisely the subgraphs $G[B_i - \{f_i\}]$ for $B_i \in \mathcal{B}_k$. Let C_k be the set of vertices of a component of $G - A''_k$ with maximum weight. Since the sets F_0, $V(G) - A''_0$, F_1, and $V(G) - A''_1$ partition the set $V(G) - A$, we have

$$\begin{aligned} &w(F_0) + \text{balance}(G - A''_0) + w(C_0) + w(F_1) + \text{balance}(G - A''_1) + w(C_1) \\ &\quad = w(F_0) + w(G - A''_0) + w(F_1) + w(G - A''_1) \\ &\quad = w(G - A) = \text{balance}(H - A) + \gamma, \end{aligned}$$

where

$$\gamma = \max_{C \in \text{Comp}(H-A)} w(C).$$

The above together with (8.9), (8.10), and (8.8) implies

$$w(C_0) + w(C_1) > c \cdot \text{balance}(H - A) + \gamma. \tag{8.11}$$

Since each C_k is contained in one component of $H - A$, we have

$$w(C_k) \leqslant \gamma \quad \text{for } k \in \{0, 1\},$$

which by (8.11) implies

$$w(C_k) > c \cdot \text{balance}(H - A) \quad \text{for } k \in \{0, 1\}. \tag{8.12}$$

If C_0 and C_1 are contained in the same component of $H - A$ then we have

$$w(C_0) + w(C_1) \leqslant \gamma,$$

which contradicts (8.11). Thus C_0 and C_1 are contained in distinct components of $H - A$. It follows that $H - (C_0 \cup C_1)$ consists of two components each containing a vertex from A. This implies that $G - (C_0 \cup C_1)$ is connected, as it contains the whole connected set A. Moreover, $G[C_0]$ and $G[C_1]$ are precisely the components of $G[C_0 \cup C_1]$, and thus we have

$$\begin{aligned}\operatorname{balance} G[C_0 \cup C_1] &= w(C_0 \cup C_1) - \max\{w(C_0), w(C_1)\} \\ &= \min\{w(C_0), w(C_1)\} \\ &> c \cdot \operatorname{balance}(H - A) \qquad \text{by (8.12)}.\end{aligned}$$

This shows that (1) is satisfied for $S = V(G) - (C_0 \cup C_1)$. □

Corollary 8.4. *There is $c \in (0, 1]$ such that for every weighted graph G containing a Hamiltonian cycle one of the following holds:*

(1) There is a connected set $S \subseteq V(G)$ such that

$$\operatorname{balance}(G - S) \geqslant c \cdot w(G).$$

(2) There is a connected set $S \subseteq V(G)$ such that

$$w(N_G(S)) \geqslant c \cdot w(G).$$

(3) There is a set $S \subseteq V(G)$ such that $G\{S\}$ is a cycle and

$$w(S) \geqslant c \cdot w(G).$$

Proof. Define $c = c' \cdot c''$, where c' and c'' are constants claimed by Lemmas 8.2 and 8.3 respectively. To see that this choice of c satisfies the conclusion, let G be a weighted graph containing a Hamiltonian cycle H. By Lemma 8.2 one of the following holds:

(1′) There is a connected set $A \subseteq V(G)$ such that

$$\operatorname{balance}(H - A) \geqslant c' \cdot w(G).$$

(2′) There is a set $S \subseteq V(G)$ such that $G\{S\}$ is a cycle and

$$w(S) \geqslant c' \cdot w(G).$$

If (2′) holds then this directly implies (3). Thus assume (1′). By Lemma 8.3 one of the following holds:

(1″) There is a connected set $S \subseteq V(G)$ such that $A \subseteq S$ and

$$\operatorname{balance}(G - S) \geqslant c'' \cdot \operatorname{balance}(H - A) \geqslant c \cdot w(G).$$

(2″) There is a connected set $S \subseteq V(G)$ such that $A \subseteq S$ and

$$w(N_G(S)) \geqslant c'' \cdot \operatorname{balance}(H - A) \geqslant c \cdot w(G).$$

Now, (1) follows from (1″), and (2) follows from (2″). □

Lemma 8.5 (Gągol [6]). *Every weighted connected graph G contains a cycle H such that every component of $G - V(H)$ has weight at most $\frac{1}{2}w(G)$.*

Proof. Let G be a weighted connected graph. Choose any $r \in V(G)$. Let T be a depth-first search tree in G with respect to r, guaranteed by Proposition 3.2. Let v be a weighted center of T, which exists by Proposition 3.1. If $v = r$ then no edge in G connects two distinct components of $T - \{v\}$ (any such edge would be a crossing edge with respect to T and r) and the conclusion follows by choosing $H = \{v\}$. Thus assume $v \neq r$. Let C be the component of $T - \{v\}$ containing r, and let $C^\star$ be the union of all other components of $T - \{v\}$. Since there are no crossing edges, all edges of G connecting two distinct components of $T - \{v\}$ join C with $C^\star$. Let xy be an edge of G connecting $x \in V(C)$ and $y \in V(C^\star)$ and minimizing $\mathrm{dist}_T(r, x)$. It follows that $x \in V(T_{rv})$ as otherwise xy would be a crossing edge. Every other edge $x'y' \in E(G)$ connecting $x' \in V(C)$ and $y' \in V(C^\star)$ also satisfies $x' \in V(T_{rv})$, and thus by minimality of $\mathrm{dist}_T(r, x)$ it satisfies $x' \in V(T_{xv})$. This shows that every component of $G - V(T_{xv})$ is entirely contained in one component of $T - \{v\}$. Let H be the cycle formed by the path T_{xy} and the edge xy. Since $V(T_{xv}) \subseteq V(H)$, every component of $G - V(H)$ is entirely contained in one component of $T - \{v\}$. Therefore, since v is a weighted center of T, every component of $G - V(H)$ has weight at most $\frac{1}{2}w(G)$. □

Corollary 8.6. *There is a constant $c \in (0, 1]$ such that for every weighted connected graph G one of the following holds:*

(1) There is a connected set $S \subseteq V(G)$ such that

$$\mathrm{balance}(G - S) \geqslant c \cdot w(G).$$

(2) There is a connected set $S \subseteq V(G)$ such that

$$w(N_G(S)) \geqslant c \cdot w(G).$$

(3) There is a set $S \subseteq V(G)$ such that $G\{S\}$ is a cycle and

$$w(S) \geqslant c \cdot w(G).$$

Proof. Let $c' \in (0, 1]$ be a constant claimed by Corollary 8.4. Set

$$c = \frac{c'}{2(1 + c')}.$$

It follows that

$$c'(\tfrac{1}{2} - c) = c. \tag{8.13}$$

We show that the choice of c above satisfies the conclusion of the lemma.

Let G be a weighted connected graph. By Lemma 8.5 there is a cycle H in G such that every component of $G - V(H)$ has weight at most $\frac{1}{2}w(G)$. If $w(H) \leqslant (\frac{1}{2} - c)w(G)$ then the conclusion (1) with $S = V(H)$ follows:

$$\mathrm{balance}(G - S) \geqslant w(G - S) - \tfrac{1}{2}w(G) = \tfrac{1}{2}w(G) - w(H) \geqslant c \cdot w(G).$$

Thus assume $w(H) > (\frac{1}{2} - c)w(G)$. Let $G' = G\{V(H)\}$. By (8.13) we have

$$c' \cdot w(G') = c' \cdot w(H) > c'(\tfrac{1}{2} - c)w(G) = c \cdot w(G).$$

By Corollary 8.4 one of the following holds:

(1′) There is a connected set $S' \subseteq V(G')$ such that

$$\mathrm{balance}(G' - S') \geqslant c' \cdot w(G') > c \cdot w(G).$$

(2′) There is a connected set $S' \subseteq V(G')$ such that

$$w(N_{G'}(S')) \geqslant c' \cdot w(G') > c \cdot w(G).$$

(3′) There is a set $S' \subseteq V(G')$ such that $G'\{S'\}$ is a cycle and

$$w(S') \geqslant c' \cdot w(G') > c \cdot w(G).$$

We show that each of the statements (1′)–(3′) above implies the corresponding statement (1)–(3) in the conclusion of the lemma.

Suppose that (1′) holds. Let S be the set of all vertices of G reachable in G by a path starting in S' and containing no other vertex of G'. Clearly, $S \cap V(G') = S'$. If uv is an edge of $G'[S']$ then the whole path from u to v in G witnessing the edge uv in G' belongs to S. Therefore, since $G'[S']$ is connected, $G[S]$ is connected too. Moreover, if two vertices from $G' - S'$ belong to distinct components of $G' - S'$, they also belong to distinct components of $G - S$, as otherwise a path connecting them in $G - S$ would witness a path connecting them in $G' - S'$. Therefore, if C is a component of $G - S$ then $V(C) \cap V(G')$ is entirely contained in one component of $G' - S'$ and hence

$$\begin{aligned} w(C) &= w\big(V(C) \cap V(G')\big) + w\big(V(C) - V(G')\big) \\ &\leqslant \max_{C' \in \mathrm{Comp}(G'-S')} w(C') + w\big(G - (S \cup V(G'))\big). \end{aligned}$$

We conclude that (1) holds:

$$\begin{aligned} \mathrm{balance}(G - S) &= w(G - S) - \max_{C \in \mathrm{Comp}(G-S)} w(C) \\ &\geqslant w(G' - S') + w\big(G - (S \cup V(G'))\big) \\ &\qquad - \max_{C' \in \mathrm{Comp}(G'-S')} w(C') - w\big(G - (S \cup V(G'))\big) \\ &= \mathrm{balance}(G' - S') > c \cdot w(G). \end{aligned}$$

Now, suppose that (2′) holds. Again, let S be the set of all vertices in G reachable in G by a path starting in S' and containing no other vertex of G'. As before, $S \cap V(G') = S'$ and $G[S]$ is connected. Moreover, if uv is an edge in G' such that $u \in S'$ and $v \in N_{G'}(S')$ then $v \in N_G(S)$ as the entire path from u to v in G inducing the edge uv in G' except v is included in S. Therefore, $N_{G'}(S') \subseteq N_G(S)$ and hence (2) follows:

$$w(N_G(S)) \geqslant w(N_{G'}(S')) > c \cdot w(G).$$

Finally, suppose that (3′) holds. Let $S = S'$. We have

$$w(S) = w(S') > c \cdot w(G).$$

Moreover, $G\{S\} = G'\{S'\}$ and hence (3) follows. □

8.2. Graphs with forbidden subdivision

The following lemma is a joint work with Adam Gągol.

Lemma 8.7. *For any $n \in \mathbb{N}^+$ and $m \in \{0, \ldots, \binom{n}{2}\}$ there is $c_{n,m} \in (0, 1]$ such that for every weighted connected graph G and every connected set $A \subseteq V(G)$ one of the following holds:*

(1) There is a connected set $S \subseteq V(G)$ such that $A \subseteq S$ and

$$w(N_G(A) - S) - \max_{C \in \mathrm{Comp}(G-S)} w\big(V(C) \cap N_G(A)\big) \geqslant c_{n,m} \cdot w(N_G(A)).$$

(2) There is a vertex $v \in N_G(A)$ such that

$$w(v) \geqslant c_{n,m} \cdot w(N_G(A)).$$

(3) There is a graph H with n vertices and m edges such that $V(H) \subseteq N_G(A)$ and $G - A$ contains a subdivision of H.

Proof. Fix $n \in \mathbb{N}^+$. The proof goes by induction on m. If $m = 0$ then setting $c_{n,0} = 1/n$ does the job. Indeed, let G be a weighted connected graph and A be a connected subset of $V(G)$. If $N_G(A)$ has at most n vertices then the heaviest of them satisfies (2). If $N_G(A)$ has at least n vertices then $G - A$ contains a subgraph H with n vertices and no edges. Thus assume $1 \leqslant m \leqslant \binom{n}{2}$. Set

$$\beta_{n,m} = \frac{c_{n,m-1}}{2(1 + c_{n,m-1})}, \tag{8.14}$$

$$c_{n,m} = \frac{\beta_{n,m}}{1 + \beta_{n,m}}. \tag{8.15}$$

We show that this choice of $c_{n,m}$ satisfies the conclusion of the lemma.

Let G be a weighted connected graph and A be a connected subset of $V(G)$. If every component C of $G - A$ satisfies

$$w\big(V(C) \cap N_G(A)\big) \leqslant (1 - c_{n,m})w(N_G(A))$$

then (2) holds for $S = A$. Thus assume that there is a component C of $G - A$ such that

$$w\big(V(C) \cap N_G(A)\big) > (1 - c_{n,m})w(N_G(A)).$$

Define $B = V(C) \cap N_G(A)$. It follows that

$$\begin{aligned} w(B) &> (1 - c_{n,m})w(N_G(A)) \\ &= \frac{1}{1 + \beta_{n,m}} w(N_G(A)) \qquad \text{by (8.15)}. \end{aligned} \tag{8.16}$$

Choose any $v_0 \in B$. For $i \in \mathbb{N}$ let B_i be the set of vertices at distance i from v_0 in $C\{B\}$. Let k be the greatest index for which $B_k \neq \emptyset$. Clearly, the sets $B_0, \ldots, B_k$ form a partition of B. Choose $j \in \{0, \ldots, k\}$ so that

$$\begin{aligned} w(B_0 \cup \ldots \cup B_{j-1}) &\leqslant \tfrac{1}{2} w(B), \\ w(B_{j+1} \cup \ldots \cup B_k) &\leqslant \tfrac{1}{2} w(B). \end{aligned} \tag{8.17}$$

Suppose

$$w(B_j) \leqslant (\tfrac{1}{2} - \beta_{n,m})w(B).$$

It follows from the above and (8.17) that

$$\begin{aligned} w(B_0 \cup \ldots \cup B_{j-1}) &= w(B) - w(B_j) - w(B_{j+1} \cup \ldots \cup B_k) \\ &\geqslant \beta_{n,m} \cdot w(B), \\ w(B_{j+1} \cup \ldots \cup B_k) &= w(B) - w(B_j) - w(B_0 \cup \ldots \cup B_{j-1}) \\ &\geqslant \beta_{n,m} \cdot w(B). \end{aligned}$$

Since there is no path in G connecting $B_0 \cup \ldots \cup B_{j-1}$ and $B_{j+1} \cup \ldots \cup B_k$ avoiding $A \cup B_j$, every component of $G - (A \cup B_j)$ is disjoint from $B_0 \cup \ldots \cup B_{j-1}$ or $B_{j+1} \cup \ldots \cup B_k$. Thus (1) follows for $S = A \cup B_j$:

$$\begin{aligned} w(N_G(A) - S) - \max_{C \in \mathrm{Comp}(G-S)} w\big(V(C) \cap N_G(A)\big) \\ \geqslant \beta_{n,m} \cdot w(B) \\ > c_{n,m} \cdot w(N_G(A)) \qquad \text{by (8.16) and (8.15).} \end{aligned}$$

It remains to consider the case

$$w(B_j) > (\tfrac{1}{2} - \beta_{n,m})w(B). \tag{8.18}$$

Every vertex in B_j is reachable in C from v_0 by a path avoiding all other vertices from B_j. Let A' be the set of vertices of G reachable in G from A by a path entirely disjoint from B_j. It follows that $A \subseteq A'$ and

$$N_G(A') = B_j = N_G(A) - A'. \tag{8.19}$$

Since $G[A]$ is connected, $G[A']$ is connected too. By the induction hypothesis one of the following holds:

(1′) There is a connected set $S \subseteq V(G)$ such that $A' \subseteq S$ and

$$w(B_j - S) - \max_{C \in \mathrm{Comp}(G-S)} w(V(C) \cap B_j) \geqslant c_{n,m-1} \cdot w(B_j).$$

(2′) There is a vertex $v \in B_j$ such that

$$w(v) \geqslant c_{n,m-1} \cdot w(B_j).$$

(3′) There is a graph H' with n vertices and $m-1$ edges such that $V(H') \subseteq B_j$ and $G - A'$ contains a subdivision of H'.

Moreover, we have

$$\begin{aligned} c_{n,m-1} \cdot w(B_j) &> \beta_{n,m} \cdot w(B) && \text{by (8.18) and (8.14)} \\ &> c_{n,m} \cdot w(N_G(A)) && \text{by (8.16) and (8.15).} \end{aligned} \tag{8.20}$$

If (1′) holds then by (8.19) we have $B_j - S = N_G(A) - S$ and $V(C) \cap B_j = V(C) \cap N_G(A)$. This and (8.20) imply (1) for the same set S. If (2′) holds then by (8.20) we have (2) for the same set S. So suppose that (3′) holds.

Let u and v be any two vertices of H' such that $uv \notin E(H')$. To prove (3) we show that $G - A$ contains a subdivision of the graph H with $V(H) = V(H')$ and $E(H) = E(H') \cup \{uv\}$. Let F' be a subdivision of H' in $G - A'$ claimed by (3′). Since $u, v \in B_j$, the vertices u and v are reachable in $G - A$ from v_0 by paths P_u and P_v, respectively, avoiding all other vertices from B_j. Let P be the path connecting u and v in $P_u \cup P_v$. It follows from the definition of A' that $V(P_u) - \{u\} \subseteq A'$ and $V(P_v) - \{v\} \subseteq A'$, and thus $V(P) - \{u, v\} \subseteq A'$. In particular, P is internally disjoint from F'. This shows that $F' \cup P$ is a subdivision of H in $G - A$. □

Corollary 8.8. *For every $n \in \mathbb{N}^+$ there is $c_n \in (0, 1]$ such that for every weighted connected graph G containing no subdivision of K_n one of the following holds:*

(1) There is a connected set $S \subseteq V(G)$ *such that*

$$\operatorname{balance}(G - S) \geqslant c_n \cdot w(G).$$

(2) There is a set $S \subseteq V(G)$ *such that* $G\{S\}$ *is a cycle and*

$$w(S) \geqslant c_n \cdot w(G).$$

Proof. Fix $n \in \mathbb{N}^+$ and set $c_n = c' \cdot c''_n$, where c' is a constant claimed by Corollary 8.6, and c''_n is a constant claimed by Lemma 8.7 for $m = \binom{n}{2}$. To see that this choice of c_n satisfies the conclusion, let G be a weighted connected graph containing no subdivision of K_n. It follows from Corollary 8.6 that (1) or (2) holds or there is a connected set $A \subseteq V(G)$ such that

$$w(N_G(A)) \geqslant c' \cdot w(G).$$

In the latter case by Lemma 8.7 one of the following holds:

(1′) There is a connected set $S \subseteq V(G)$ such that $A \subseteq S$ and

$$w(N_G(A) - S) - \max_{C \in \operatorname{Comp}(G-S)} w\big(V(C) \cap N_G(A)\big) \geqslant c''_n \cdot w(N_G(A)).$$

(2′) There is a vertex $v \in N_G(A)$ such that

$$w(v) \geqslant c''_n \cdot w(N_G(A)).$$

The third case from Lemma 8.7 is excluded by the assumption that G contains no subdivision of K_n. If (1′) holds then every component C of $G - S$ satisfies

$$\begin{aligned} w(C) &= w\big(V(C) \cap N_G(A)\big) + w\big(V(C) - N_G(A)\big) \\ &\leqslant w\big(V(C) \cap N_G(A)\big) + w\big(G - (S \cup N_G(A))\big), \end{aligned}$$

and hence (1) follows:

$$\begin{aligned} \operatorname{balance}(G - S) &= w(G - S) - \max_{C \in \operatorname{Comp}(G-S)} w(C) \\ &\geqslant w(N_G(A) - S) + w\big(G - (S \cup N_G(A))\big) \\ &\quad - \max_{C \in \operatorname{Comp}(G-S)} w\big(V(C) \cap N_G(A)\big) - w\big(G - (S \cup N_G(A))\big) \\ &\geqslant c''_n \cdot w(N_G(A)) \\ &\geqslant c_n \cdot w(G). \end{aligned}$$

If (2′) holds then (2) follows for $S = \{v\}$. □

Chapter 9

Game T on odd graphs with forbidden subdivision

In this chapter we prove Theorem 1.3, namely, that for a suitable constant $c_n > 0$ Alice can secure at least $c_n \cdot w(G)$ in the game $\mathsf{T}(G)$ for any weighted connected graph G with an odd number of vertices and no subdivision of K_n.

We denote by $\operatorname{supp} G$ the set of vertices of a weighted graph G that have positive weight. We call a set $S \subseteq V(G)$ *sparse* if the distance in G between any two vertices from S is at least 3. Equivalently, S is sparse if the closed neighborhoods in G of the vertices from S are pairwise disjoint. We call G *sparsely weighted* if $\operatorname{supp} G$ is sparse.

First we prove that Alice has a strategy in $\mathsf{T}(G)$ to gather at least $c_n \cdot w(G)$ for any sparsely weighted connected graph G with an odd number of vertices and with no subdivision of K_n. This strategy can be as well applied to graphs that contain a sparse set of vertices with substantial weight (at least a suitable constant fraction of $w(G)$). Then, to prove the theorem for all weighted connected graphs with an odd number of vertices and no subdivision of K_n, we present a complementary strategy of Alice for graphs containing no sparse set of vertices with substantial weight.

9.1. Strategies for sparsely weighted graphs

For this entire section we assume that G is a sparsely weighted connected graph with an odd number of vertices. It follows that the family

$$\big\{N_G[v] : v \in \operatorname{supp} G\big\} \cup \big\{\{v\} : v \in V(G) - N_G[\operatorname{supp} G]\big\}$$

is a partition of $V(G)$ into sets of radius 1 or 0. We define

$$G^R = G/\big(\big\{N_G[v] : v \in \operatorname{supp} G\big\} \cup \big\{\{v\} : v \in V(G) - N_G[\operatorname{supp} G]\big\}\big).$$

Thus G^R is a 1-shallow minor of G. It is weighted with weight function w inherited from G. In particular, we have $w(G^R) = w(G)$.

Lemma 9.1. *For every connected set $S^R \subseteq V(G^R)$ Alice has a strategy in $\mathsf{T}(G)$ to collect vertices of total weight at least $\frac{1}{2}\operatorname{balance}(G^R - S^R)$.*

Proof. Let S^R be a connected subset of $V(G^R)$. Let $S = \bigcup S^R$. Clearly, S is connected in G and disjoint from $N_G[\operatorname{supp} G - S]$.

Alice starts by taking any vertex from S. If Bob takes a vertex from $N(v)$ for some $v \in \operatorname{supp} G - S$, Alice answers by taking v. Otherwise, unless the entire S is taken, Alice picks a next available vertex from S. Consider Alice's first turn at which the entire S is taken. Let T be the set of vertices taken so far. Thus $S \subseteq T$. Since all vertices from $(T - S) \cap \operatorname{supp} G$ are taken by Alice, her partial outcome is at least $w(T - S)$. The subgame of $\mathsf{T}(G)$ at the considered position is $\mathsf{Tr}(G - T)$, where $G - T$ is considered as a rooted graph with root set $N_G(T)$. We apply a strategy claimed by Corollary 4.4 to this subgame. It follows that Alice's total outcome is at least

$$\begin{aligned} w(T-S) + \tfrac{1}{2}w(G-T) - \tfrac{1}{2}\max_{C\in\mathrm{Comp}(G-T)} w(C) \\ \geqslant \tfrac{1}{2}w(G-S) - \tfrac{1}{2}\max_{C\in\mathrm{Comp}(G-S)} w(C) \\ = \tfrac{1}{2}w(G^R - S^R) - \tfrac{1}{2}\max_{C^R\in\mathrm{Comp}(G^R-S^R)} w(C^R) \\ = \tfrac{1}{2}\,\mathrm{balance}(G^R - S^R). \end{aligned}$$ □

Lemma 9.2. *For every set $S^R \subseteq V(G^R)$ such that $G^R\{S^R\}$ is a cycle Alice has a strategy in $\mathsf{T}(G)$ to collect vertices of total weight at least $\frac{1}{6}w(S^R)$.*

Proof. Let S^R be a subset of $V(G^R)$ such that $G^R\{S^R\}$ is a cycle. For $v \in \operatorname{supp} G$ define $v^R = N_G[v]$. Without loss of generality assume that S^R consists only of vertices of the form v^R for $v \in \operatorname{supp} G$. Define

$$S = \{v \in \operatorname{supp} G : v^R \in S^R\}.$$

Let $n = |S| = |S^R|$. Enumerate the vertices in S as $v_0, \dots, v_{n-1}$ so that $v_0^R, \dots, v_{n-1}^R$ occur in this order on the cycle $G^R\{S^R\}$. Define $v_n = v_0$. Since $G^R\{S^R\}$ is a cycle, the neighborhood of every component of $G^R - S^R$ consists of a single vertex v_i^R or two consecutive vertices v_i^R and v_{i+1}^R. Thus the neighborhood of every component of $G - N_G[S]$ in G is adjacent to either one set $N_G(v_i)$ or two consecutive sets $N_G(v_i)$ and $N_G(v_{i+1})$. Let A_i denote the union of $N_G[v_i]$ and all components of $G - N_G[S]$ adjacent only to $N_G(v_i)$. Let $A_{i,i+1}$ denote the union of all components of $G - N_G[S]$ adjacent to $N_G(v_i)$ and $N_G(v_{i+1})$. The sets A_i and $A_{i,i+1}$ together form a partition of $V(G)$. For $0 \leqslant i \leqslant j \leqslant n-1$ define

$$\begin{aligned} A(i,j) &= A_{i,i+1} \cup A_{i+1} \cup A_{i+1,i+2} \cup \ldots \cup A_{j-1} \cup A_{j-1,j}, \\ A[i,j) &= A_i \cup A_{i,i+1} \cup A_{i+1} \cup A_{i+1,i+2} \cup \ldots \cup A_{j-1} \cup A_{j-1,j}, \\ A[i,j] &= A_i \cup A_{i,i+1} \cup A_{i+1} \cup A_{i+1,i+2} \cup \ldots \cup A_{j-1} \cup A_{j-1,j} \cup A_j. \end{aligned}$$

Define

$$\begin{aligned} C_0 &= \{v_i \in S : |A[0,i)| \text{ is even}\}, \\ C_1 &= \{v_i \in S : |A[0,i)| \text{ is odd}\}. \end{aligned}$$

Thus $C_0 \cup C_1 = S$. Choose $C = C_0$ or $C = C_1$ so that $w(C) \geqslant \frac{1}{2}w(S)$. It follows from the above definitions that for $0 \leqslant i < j \leqslant n-1$ if $v_i, v_j \in C$ then $|A[i,j)|$ is even. Define

$$\begin{aligned} S_0 &= \{v_i \in S : |A_i| \text{ is even}\}, \\ S_1 &= \{v_i \in S : |A_i| \text{ is odd}\}. \end{aligned}$$

Thus $S_0 \cup S_1 = S$. We prove the following two claims:

(1) Alice has a strategy in $\mathsf{T}(G)$ to secure at least $w(S_0 \cap C)$.
(2) Alice has a strategy in $\mathsf{T}(G)$ to secure at least $\frac{1}{2}w(S_1 \cap C)$.

This suffices for the conclusion of the lemma: if $w(S_0 \cap C) \geqslant \frac{1}{3}w(C)$ then Alice can choose a strategy claimed by (1), while if $w(S_1 \cap C) \geqslant \frac{2}{3}w(C)$ then she can choose a strategy claimed by (2).

First we present a strategy for Alice claimed by (1). She starts by taking v_0. Then she sticks to the following two rules at each her turn:

- Always take a vertex in S if available.
- Never take a vertex in $N_G(v_i)$ for an untaken $v_i \in S$ unless forced to.

The first rule ensures that the taken vertices from S always form an interval in the cyclic order on S. Suppose that at some point of the game Alice is forced to take a vertex from $N_G(v_i)$ for some untaken vertex $v_i \in S_0 \cap C$. It follows that the set of untaken vertices is of the form $A[\ell, r]$ for some ℓ and r with $1 \leqslant \ell \leqslant r \leqslant n-1$ and $v_\ell, v_r \in S_0 \cap C$. Since $v_\ell, v_r \in C$, $|A[\ell, r)|$ is even. Since $v_r \in S_0$, $|A_r|$ is even. Hence $|A[\ell, r]|$ is even too. On the other hand, since $A[\ell, r]$ is the set of taken vertices, Alice is to move, and G has an odd number of vertices, we have $|A[\ell, r)|$ odd. This contradiction shows that Alice is never forced to take a vertex from $N_G(v_i)$ for any untaken $v_i \in S_0 \cap C$, and thus Bob never gets the opportunity to take a vertex from $S_0 \cap C$. Therefore, Alice gathers the whole $S_0 \cap C$.

Now we present Alice's strategy claimed by (2). Choose an index $j \in \{0, \ldots, n-1\}$ so that

$$\begin{aligned} w\bigl(\{v_0, \ldots, v_j\} \cap S_1 \cap C\bigr) &\geqslant \tfrac{1}{2} w(S_1 \cap C), \\ w\bigl(\{v_j, \ldots, v_{n-1}\} \cap S_1 \cap C\bigr) &\geqslant \tfrac{1}{2} w(S_1 \cap C). \end{aligned} \tag{9.1}$$

Alice starts by taking v_j. Then at each her turn she obeys the same two rules as before. Again, by the first rule the taken vertices from S form an interval in the cyclic order on S. Consider the first position in the game at which Alice is forced to take a vertex from $N_G(v_i)$ for some untaken vertex $v_i \in S_1 \cap C$. This is Alice's first turn after which Bob has the opportunity to take a vertex from $S_1 \cap C$. Suppose that v_0 and v_{n-1} are not taken yet. Let ℓ and r be such that $0 \leqslant \ell < r \leqslant n-1$ and $v_{\ell+1}, \ldots, v_{r-1}$ are the taken vertices from S. Thus $v_\ell, v_r \in S_1 \cap C$, and the set of taken vertices is precisely $A(\ell, r)$. Since $v_\ell, v_r \in C$, $|A[\ell, r)|$ is even. Since $v_\ell \in S_1$, $|A_\ell|$ is odd. Hence $|A(\ell, r)|$ is odd. On the other hand, since $A(\ell, r)$ is the set of taken vertices and Alice is to move, $|A(\ell, r)|$ is even. This contradiction shows that at least one of v_1 and v_n is already taken and thus all $v_0, \ldots, v_j$ or all $v_j, \ldots, v_{n-1}$ are taken. Since all vertices from $S_1 \cap C$ taken so far have been taken by Alice, by (9.1) her partial outcome is at least $\frac{1}{2} w(S_1 \cap C)$. If the considered position never occurs, Alice gathers the whole $S_1 \cap C$. □

Corollary 9.3. *For every $n \in \mathbb{N}^+$ there is $c_n \in (0, 1]$ such that if G contains no subdivision of K_n then Alice has a strategy in $\mathsf{T}(G)$ to collect vertices of total weight at least $c_n \cdot w(G)$.*

Proof. Fix $n \in \mathbb{N}^+$. By Corollary 3.6 there is $N \in \mathbb{N}^+$ such that if G contains no subdivision of K_n then G^R contains no subdivision of K_N. By Corollary 8.8 there is $c'_N \in (0, 1]$ such that if G^R contains no subdivision of K_N then one of the following holds:

(1) There is a connected set $S^R \subseteq V(G^R)$ such that
$$\operatorname{balance}(G^R - S^R) \geqslant c'_N \cdot w(G^R).$$
(2) There is a set $S^R \subseteq V(G^R)$ such that $G^R\{S^R\}$ is a cycle and
$$w(S^R) \geqslant c'_N \cdot w(G^R).$$

If (1) holds then by Lemma 9.1 Alice has a strategy in $\mathsf{T}(G)$ to collect vertices of total weight at least

$$\tfrac{1}{2} \operatorname{balance}(G^R - S^R) \geqslant \tfrac{1}{2} c'_N \cdot w(G^R).$$

If (2) holds then by Lemma 9.2 Alice has a strategy in $\mathsf{T}(G)$ to collect vertices of total weight at least

$$\tfrac{1}{6}w(S^R) \geqslant \tfrac{1}{6}c'_N \cdot w(G^R).$$

Thus setting $c_n = \frac{1}{6}c'_N$ yields the conclusion. □

9.2. Complementary strategy for general graphs

Recall that for a linear ordering σ of a set X and for $x \in X$ we define

$$\sigma^-(x) = \{y \in X : y <_\sigma x\}.$$

Let G be a weighted connected graph with at least two vertices. A subset I of $V(G)$ is called *independent* if no two vertices from I are adjacent in G. Let I be an independent subset of $V(G)$. A linear ordering σ of $V(G)-I$ is called *legal* if every vertex $v \in V(G)-I$ except the least one is adjacent to

$$\sigma^-(v) \cup \big(N_G(\sigma^-(v)) \cap I\big).$$

Let σ be a legal linear ordering of $V(G)-I$. For $v \in V(G)-I$ define

$$B_\sigma(v) = \big(N_G(v) \cap I\big) - \big(N_G(\sigma^-(v)) \cap I\big).$$

Note that all non-empty sets $B_\sigma(v)$ form a partition of I. If $B_\sigma(v) \neq \emptyset$ then define $u_\sigma(v)$ to be a vertex from $B_\sigma(v)$ of maximum weight. Define

$$U_\sigma = \{u_\sigma(v) : v \in V(G)-I \text{ and } B_\sigma(v) \neq \emptyset\}.$$

Note that $U_\sigma \subseteq I$.

Lemma 9.4. *For every weighted connected graph G, every independent set $I \subseteq V(G)$, and every legal linear ordering σ of $V(G)-I$, Alice has a strategy in $\mathsf{T}(G)$ to collect vertices of total weight at least $\frac{1}{2}w(I-U_\sigma)$.*

Proof. Let G be a weighted connected graph with at least two vertices, I be an independent subset of $V(G)$, and σ be a legal linear ordering of $V(G)-I$. We can assume without loss of generality that only vertices in I have positive weight. The strategy we construct for Alice is as follows. Start by taking the least vertex in $V(G)-I$ with respect to σ. In every subsequent move, if a vertex from I is available then take one with maximum weight, otherwise take the least untaken vertex $v \in V(G)-I$ with respect to σ, which is available by the property of σ being legal.

It suffices to show that Bob's final outcome is at most Alice's outcome plus $w(U_\sigma)$. To this end we bound the weight of every vertex $u \in I$ collected by Bob from above by $w(v)$ or $w(u_\sigma(v))$, where v is the vertex taken by Alice in her directly preceding move. Consider the position in the game just before Alice takes v. If $v \in I$ then both u and v are available at this position and thus $w(u) \leqslant w(v)$. Otherwise the move of Alice taking v makes u available and hence $u \in B_\sigma(v)$, which implies $w(u) \leqslant w(u_\sigma(v))$. □

Theorem 9.5. *For every $n \in \mathbb{N}^+$ there is $c_n \in (0,1]$ such that for every weighted connected graph G with an odd number of vertices containing no subdivision of K_n Alice has a strategy in $\mathsf{T}(G)$ to collect vertices of total weight at least $c_n \cdot w(G)$.*

Proof. Fix $n \in \mathbb{N}^+$. Let d be an upper bound on $\operatorname{wcol}_2 G$ claimed by Theorem 3.8 for graphs G containing no subdivision of K_n. Let $c_n' \in (0,1]$ be a constant claimed by Corollary 9.3. Set

$$c_n = \frac{c_n'}{d^2 + 2d(d^2 - d + 1)c_n'}. \tag{9.2}$$

Let G be a weighted connected graph with an odd number of vertices containing no subdivision of K_n. Let π be a linear ordering of the vertices of G realizing $\operatorname{wcol}_2 G$. Since $\operatorname{wcol}_2 G \leqslant d$, Proposition 3.7 provides a coloring of the vertices of G using at most d colors so that no vertex is weakly 2-accessible from another vertex of the same color. Select a color class I with maximum total weight. Clearly

$$w(I) \geqslant \frac{w(G)}{d}. \tag{9.3}$$

The property of the coloring implies that I is independent and every two distinct vertices $u, v \in I$ satisfy $N_G(u) \cap N_G(v) \subseteq \pi^-(u) \cap \pi^-(v)$.

Let σ be a legal linear ordering of $V(G) - I$ minimizing $w(U_\sigma)$. If

$$w(U_\sigma) \leqslant (1 - 2dc_n)w(I)$$

then by Lemma 9.4 Alice has a strategy in $\mathsf{T}(G)$ to collect vertices of total weight at least

$$\begin{aligned} \tfrac{1}{2}w(I - U_\sigma) &\geqslant dc_n \cdot w(I) \\ &\geqslant c_n \cdot w(G) \quad \text{by (9.3)}. \end{aligned}$$

Thus assume

$$w(U_\sigma) > (1 - 2dc_n)w(I). \tag{9.4}$$

Assign each vertex $v \in I - U_\sigma$ a potential $\phi(v)$ setting its initial value as

$$\phi(v) = (d^2 - d)w(v). \tag{9.5}$$

The initial total potential is given by

$$\begin{aligned} \Phi &= \sum_{v \in I - U_\sigma} \phi(v) \\ &= (d^2 - d)w(I - U_\sigma) \quad \text{by (9.5)} \\ &< 2d(d^2 - d)c_n \cdot w(I) \quad \text{by (9.4)}. \end{aligned} \tag{9.6}$$

Set initially $S = U_\sigma$.

Here is an outline of what happens next in the proof. We remove vertices from S until every vertex in $V(G) - I$ has at most two neighbors in S. We compensate the weight of the removed vertices by the decrease of the potentials. On the other hand, we keep the potential of every vertex non-negative. Hence the initial total potential Φ is an upper bound on the total weight of all vertices being removed from S. Our procedure of removing vertices from S repeatedly selects a vertex in $V(G) - I$ with more than two neighbors in S and removes all these neighbors except at most two.

Let v be a vertex in $V(G) - I$ with more than two neighbors in S. Let v_0 be the least vertex with respect to σ such that v and v_0 share a neighbor in I. We have $v_0 <_\sigma v$, as otherwise v would have only one neighbor in U_σ and thus in S. Let σ' be the linear ordering of $V(G) - I$ obtained from σ by moving v to the first position greater than v_0. Clearly σ' is legal. Moreover, if $x \notin I \cup \{v\}$ then $B_{\sigma'}(x) \subseteq B_\sigma(x)$ and $B_\sigma(x) - B_{\sigma'}(x) \subseteq N_G(v)$. Define

$$X_v = \{x \in V(G) - (I \cup \{v\}) : B_\sigma(x) \neq \emptyset \text{ and } u_\sigma(x) \in N_G(v)\}.$$

We show that the partial functions $u_\sigma, u_{\sigma'} : (V(G) - I) \to I$ can differ only on vertices from $X_v \cup \{v\}$. Suppose that u_σ and $u_{\sigma'}$ differ on a vertex $x \notin I \cup \{v\}$. In particular, at least one of $u_\sigma(x)$ and $u_{\sigma'}(x)$ is defined. Since $B_{\sigma'}(x) \subseteq B_\sigma(x)$, it follows that $u_\sigma(x)$ is defined and $u_\sigma(x) \in B_\sigma(x) - B_{\sigma'}(x) \subseteq N_G(v)$. Thus $x \in X_v$.

The vertices in $U_\sigma - U_{\sigma'}$ that are still in S are the ones that we remove from S at this step of our procedure. The definition of σ' yields $B_{\sigma'}(v) = (N_G(v) \cap I) - (N_G(v_0) \cap I)$. It follows that no neighbor of v other than $u_{\sigma'}(v_0)$ or $u_{\sigma'}(v)$ belongs to $U_{\sigma'}$, so these are the only two neighbors of v that may remain in S. We also decrease the potential $\phi(u)$ of each vertex $u \in U_{\sigma'} - U_\sigma$ by $w(u)$. The difference between the total decrease of the potentials and the total weight of the vertices being removed from S at this step is at least

$$w(U_{\sigma'} - U_\sigma) - w(U_\sigma - U_{\sigma'}) = w(U_{\sigma'}) - w(U_\sigma) \geqslant 0,$$

where the last inequality follows from the choice of σ minimizing $w(U_\sigma)$.

Now we show that after performing all steps the potential $\phi(u)$ of each vertex $u \in I - U_\sigma$ remains non-negative, that is, it is decreased by $w(u)$ at most $d^2 - d$ times. If $\phi(u)$ is decreased when processing $v \in V(G) - I$ then either $u = u_{\sigma'}(v) \neq u_\sigma(v)$ or $u = u_{\sigma'}(x) \neq u_\sigma(x)$ for a vertex $x \in X_v$. The former case occurs for at most $d - 1$ choices of v, as $u_\sigma(v)$ and $u_{\sigma'}(v)$ being two vertices from $N_G(v) \cap I$ are greater than v with respect to π. The latter case occurs for at most $d - 1$ choices of x (for the same reason), each giving rise to at most $d - 1$ choices of v, as again $u_\sigma(x)$ and $u_\sigma(v)$ being two distinct vertices from $N_G(v) \cap I$ are greater than v with respect to π. Therefore, $\phi(u)$ is decreased for at most $d^2 - d$ choices of v in total.

Let $\tilde{S}$ denote the final set S after all removals. Since the initial sum of potentials Φ constitutes an upper bound for the total weight of all vertices removed from S, we have

$$\begin{aligned} w(\tilde{S}) &\geqslant w(U_\sigma) - \Phi \\ &> \big(1 - 2dc_n - 2d(d^2 - d)c_n\big)w(I) && \text{by (9.4) and (9.6)} \\ &= \frac{d^2 c_n}{c'_n} w(I) && \text{by (9.2)} \\ &\geqslant \frac{dc_n}{c'_n} w(G) && \text{by (9.3).} \end{aligned} \tag{9.7}$$

Fix a vertex $v \in \tilde{S}$. If a vertex $u \in \tilde{S}$ is at distance 2 from v in G then u and v share a neighbor $x_u \in V(G) - I$. By the definition of I and the fact that $\tilde{S} \subseteq U_\sigma \subseteq I$, neither u nor v is weakly 2-accessible from the other with respect to π. Thus $x_u <_\pi v$. By the assumption that π realizes $\operatorname{wcol}_2 G$, there are at most $d - 1$ candidates for x_u. Moreover, since every vertex in $V(G) - I$ is adjacent to at most two vertices from $\tilde{S}$, no two choices of u can give the same x_u. This shows that v is at distance 2 from at most $d - 1$ other vertices from $\tilde{S}$.

It follows that the vertices from $\tilde{S}$ can be colored using at most d colors so that no two vertices at distance 2 in G receive the same color. Let C be a color class with maximum weight. It follows that C is sparse in G and

$$\begin{aligned} w(C) &\geqslant \frac{w(\tilde{S})}{d} \\ &> \frac{c_n}{c'_n} w(G) \quad \text{by (9.7)}. \end{aligned} \tag{9.8}$$

Define G' to be the graph obtained from G by resetting the weights of all vertices in $V(G) - C$ to zero. Thus $w(G') = w(C)$. By Corollary 9.3 Alice has a strategy in $\mathsf{T}(G')$ to collect vertices of total weight at least $c'_n \cdot w(G')$, which is more than $c_n \cdot w(G)$ by (9.8). The same strategy applied to G gives Alice more than $c_n \cdot w(G)$ in $\mathsf{T}(G)$. $\square$

Chapter 10

Summary and open problems

In our study of graph sharing games we have focused on two types of questions: one concerning lower bounds on Alice's guaranteed outcome for various classes of graphs, and the other concerning efficient algorithms computing optimal strategies for the class of trees.

We showed that Alice's guaranteed outcome in both game T and game R on a tree may be arbitrarily small compared to the total weight, unless we restrict the parity of the number of vertices of the tree. We proved that in game T on a tree with an odd number of vertices Alice can secure at least $\frac{1}{4}$ of the total weight. On the other hand, there is a 9-vertex tree on which Alice can gather no more than $\frac{2}{5}$ of the total weight in game T if Bob plays optimally.

Problem 10.1. What is the best constant c such that Alice can secure at least c of the total weight of any tree with an odd number of vertices in game T?

We proved that actually, for every n, Alice can secure a positive fraction (depending only on n) of the total weight in game T on graphs with an odd number of vertices containing no subdivision of K_n. On the other hand, graphs with an odd number of vertices and bounded expansion can be arbitrarily bad for Alice. This shows that the result about forbidden subdivisions is essentially best possible with respect to the class of graphs under consideration.

The best Alice can secure on every tree with an odd number of vertices in game R is $\frac{1}{2}$ of the total weight. Beyond trees game R becomes much more complex, and we are unable to point out any natural class of graphs broader than the class of even trees on which Alice can secure a positive fraction of the total weight. The difficulty comes from the fact that the availability of a vertex may change many times during the game, whereas in game R on a tree or game Tr (and thus game T except for the first move) on any graph a vertex that once became available remains available until taken. What we know is that forbidding K_n as a subgraph is insufficient to guarantee Alice positive fraction of the total weight. The class of bipartite graphs with an even number of vertices seems to be a natural candidate to study.

Problem 10.2. Is there a constant $c > 0$ such that Alice can secure at least c of the total weight of any bipartite graph with an even number of vertices in game R?

Finally, we provided a precise description of optimal strategies of both players in game T on trees, and presented a polynomial-time algorithm computing these strategies. Whether such an algorithm exists for game R on trees remains open.

Problem 10.3. Is there a polynomial-time algorithm computing optimal strategies of both players in game R on trees?

Bibliography

[1] B. Bollobás, A. Thomason, Proof of a conjecture of Mader, Erdős and Hajnal on topological complete subgraphs, *Europ. J. Combin.* 19 (1998), 883–887.

[2] J. Cibulka, J. Kynčl, V. Mészáros, R. Stolař, P. Valtr, Solution to Peter Winkler's pizza problem, in: G. O.H. Katona, A. Schrijver, T. Szőnyi (Eds.), Fete of Combinatorics and Computer Science, Vol. 20 of Bolyai Society Mathematical Studies, Springer, 2010, pp. 69–93.

[3] J. Cibulka, J. Kynčl, V. Mészáros, R. Stolař, P. Valtr, Graph sharing games: complexity and connectivity, *Theor. Comput. Sci.*, to appear.

[4] R. Diestel, Graph theory, Third edition, Vol. 173 of Graduate Texts in Mathematics, Springer, 2005.

[5] Z. Dvořák, On forbidden subdivision characterizations of graph classes, *Europ. J. Combin.* 29 (2008), 1321–1332.

[6] A. Gągol, Parity and minors in graph sharing games, M.Sc. Thesis, Jagiellonian University, Kraków, 2011.

[7] H. A. Jung, Zusammenzüge und Unterteilungen von Graphen, *Math. Nachr.* 35 (1967), 241–267.

[8] H. A. Kierstead, private communication, 2008.

[9] H. A. Kierstead, D. Yang, Orderings on graphs and game coloring number, *Order* 20 (2003), 255–264.

[10] K. Knauer, P. Micek, T. Ueckerdt, How to eat $\frac{4}{9}$ of a pizza, *Discrete Math.* 311 (2011), 1635–1645.

[11] J. Komlós, E. Szemerédi, Topological cliques in graphs II, *Combin. Prob. Comput.* 5 (1996), 79–90.

[12] W. Mader, Homomorphieeigenschaften und mittlere Kantendichte von Graphen, *Math. Ann.* 174 (1967), 265–268.

[13] P. Micek, B. Walczak, A graph-grabbing game, *Combin. Prob. Comput.* 20 (2011), 623–629.

[14] P. Micek, B. Walczak, Parity in graph sharing games, *Discrete Math.* 312 (2012), 1788–1795.

[15] M. Rosenfeld, A gold-grabbing game, 2009, `http://garden.irmacs.sfu.ca/?q=op/a_gold_grabbing_game`.

[16] D. E. Seacrest, T. Seacrest, Grabbing the gold, *Discrete Math.* 312 (2012), 1804–1806.

[17] P. Valtr, private communication, 2009.

[18] P. M. Winkler, problem posed at Building Bridges, a conference in honor of the 60th birthday of László Lovász, Budapest, 2008.