

DISSERTATION ZUR ERLANGUNG DES DOKTORGRADES (DR. RER. NAT.) DES
FACHBEREICHS MATHEMATIK/INFORMATIK DER UNIVERSITÄT OSNABRÜCK

Cut Problems on Graphs

vorgelegt von
Alexander Nover

Osnabrück, 2022

Datum der Disputation: 06.05.2022

Prüfungskommission: Prof. Dr. Martina Juhnke-Kubitzke (Universität Osnabrück)
Prof. Dr. Markus Chimani (Universität Osnabrück)
Prof. Dr. Volker Kaibel (Otto-von-Guericke-Universität Magdeburg)
Dr. Friedrich Bökler (Universität Osnabrück)

Abstract

Cut problems on graphs are a well-known and intensively studied class of optimization problems. In this thesis, we study the maximum cut problem (MAXCUT), the maximum bond problem (MAXBOND), and the minimum multicut problem (MINMULTICUT) through their associated polyhedra, i.e., the cut polytope, the bond polytope, and the multicut dominant, respectively.

Continuing the research on MAXCUT and the cut polytope, we present a linear description for cut polytopes of $K_{3,3}$ -minor-free graphs as well as an algorithm solving MAXCUT on these graphs in the same running time as planar MAXCUT. Moreover, we give a complete characterization of simple and simplicial cut polytopes. Considering MAXCUT from an algorithmic point of view, we propose an FPT-algorithm for MAXCUT parameterized by the crossing number.

We start the structural study of the bond polytope by investigating its basic properties and the effect of graph operations on the bond polytope and its facet-defining inequalities. After presenting a linear-time reduction of MAXBOND to MAXBOND on 3-connected graphs, we discuss valid and facet defining inequalities arising from edges and cycles. These inequalities yield a complete linear description for bond polytopes of 3-connected planar $(K_5 - e)$ -minor-free graphs. This polytopal result is mirrored algorithmically by proposing a linear-time algorithm for MAXBOND on $(K_5 - e)$ -minor-free graphs.

Finally, we start the structural study of the multicut dominant. We investigate basic properties, which gives rise to lifting and projection results for the multicut dominant. Then, we study the effect of graph operations on the multicut dominant and its facet-defining inequalities. Moreover, we present facet-defining inequalities supported on stars, trees, and cycles as well as separation algorithms for the former two classes of inequalities.

Kurzzusammenfassung

Cutprobleme auf Graphen sind eine bekannte und intensiv untersuchte Klasse von Optimierungsproblemen. In dieser Arbeit untersuchen wir das Maximum-Cut-Problem (MAXCUT), das Maximum-Bond-Problem (MAXBOND) und das Minimum-Multicut-Problem (MINMULTICUT) über ihre jeweiligen assoziierten Polyeder, d.h. das Cut-Polytop, das Bond-Polytop und die Multicut-Dominante.

Als Fortführung der Forschung zu MAXCUT präsentieren wir eine lineare Beschreibung für Cut-Polytope von $K_{3,3}$ -Minor-freien Graphen sowie einen Algorithmus, der MAXCUT auf diesen Graphen in der Laufzeit von planarem MAXCUT löst. Darüber hinaus geben wir eine vollständige Charakterisierung von einfachen und simplizialen Cut-Polytopen. In einer algorithmischen Betrachtung von MAXCUT präsentieren wir einen FPT-Algorithmus für MAXCUT, der durch die Kreuzungszahl parametrisiert ist.

Wir beginnen die strukturelle Untersuchung des Bond-Polytops, wobei wir seine grundlegenden Eigenschaften und den Effekt von Graphoperationen auf das Bond-Polytop und seine facetten-definierenden Ungleichungen untersuchen. Nachdem wir eine Linearzeitreduktion von MAXBOND auf MAXBOND auf 3-zusammenhängenden Graphen vorgestellt haben, diskutieren wir gültige und facetten-definierende Ungleichungen, die sich aus Kanten und Kreisen ergeben. Diese Ungleichungen liefern eine vollständige lineare Beschreibung für Bond-Polytope von 3-zusammenhängenden, planaren, $(K_5 - e)$ -Minor-freien Graphen. Dieses polytopale Ergebnis spiegeln wir algorithmisch, indem wir einen Linearzeitalgorithmus für MAXBOND auf $(K_5 - e)$ -Minor-freien Graphen angeben.

Zuletzt beginnen wir die strukturelle Untersuchung der Multicut-Dominante. Wir untersuchen grundlegende Eigenschaften, die zu Lifting- und Projektionsergebnissen führen. Dann untersuchen wir den Effekt von Graphoperationen auf die Multicut-Dominante und ihre facetten-definierenden Ungleichungen. Darüber hinaus präsentieren wir facetten-definierende Ungleichungen, die auf Sternen, Bäumen und Kreisen basieren, sowie Separationsalgorithmen für die beiden erstgenannten Ungleichungsklassen.

Acknowledgements

First I would like to thank my supervisors Martina Juhnke-Kubitzke and Markus Chimani for giving me the opportunity to work on this thesis, for drawing my attention towards the interesting topic of polyhedra arising from optimization problems by linear programming, for constant guidance, and for being a source of motivation and support during the work on this thesis.

I thank all former and current colleagues for every mathematical and non-mathematical discussion during the coffee breaks and the overall pleasant atmosphere. I am grateful for everyone who did their best to make the work as pleasant and efficient as possible in times of corona lockdowns and home offices. In particular, I would like to thank Niklas Troost for proofreading parts of this thesis.

Last but not least, I would like to thank my family and friends for the support, encouragement, and distraction whenever I needed it over the past years.

Contents

Introduction	1
1 Preliminaries	5
1.1 Graphs	5
1.2 Polyhedra	6
1.3 Integer Linear Programming	8
1.4 The Maximum Cut Problem	10
1.5 Cut Polytopes	10
2 Cut Polytopes of Minor-free Graphs	17
2.1 $K_{3,3}$ -minor-free Graphs	18
2.2 Simple and Simplicial Cut Polytopes	24
2.3 Open Problems	26
3 Maximum Cut Parameterized by Crossing Number	29
3.1 Preliminaries	30
3.2 Algorithm	31
3.3 Minor Crossing Number	37
3.4 Open Problems	38
4 On the Bond Polytope	41
4.1 First Properties and Comparison to $\text{CUT}^\square(G)$	43
4.2 Constructing Facets from Facets	46
4.3 Reduction to 3-connectivity	51
4.4 Non-Interleaved Cycle Inequalities	53
4.5 Edge- and Interleaved Cycle Inequalities	58
4.6 $(K_5 - e)$ -Minor-Free Graphs	62
4.7 Open Problems	65

5	On the Dominant of the Multicut Polytope	67
5.1	Basic Properties	69
5.2	Constructing Facets from Facets	74
5.3	Star Inequalities	78
5.4	Tree Inequalities	83
5.5	Cycle Inequalities	87
5.6	Open Problems	91
6	Conclusion	93
	Bibliography	95

Introduction

Cut problems in graphs are a well-established class of problems attracting interest since the beginning of modern algorithmic research. Probably the most prominent such problems are the maximum cut problem and the minimum cut problem. Given a graph $G = (V, E)$, a selection of nodes $S \subseteq V$ defines a *cut* in G as an edge set $\delta_G(S) = \{e \in E : |e \cap S| = 1\}$.

The *maximum cut problem* in a weighted graph, MAXCUT for short, is well-known in combinatorial optimization, and one of Karp's original 21 NP-complete problems [Kar72]. Formally, considering a graph G with edge weights $c_e \in \mathbb{R}$, MAXCUT is the problem of finding a cut $\delta_G(S)$ that maximizes $\sum_{e \in \delta_G(S)} c_e$. Although MAXCUT is NP-complete in general, polynomial algorithms are known for some graph classes, e.g., planar graphs or more generally K_5 -minor-free graphs [Bar83, OD72, Had75] (see Chapter 1.4 for more details).

The problem is constantly receiving attention in the literature due to its applicability to various scenarios: these range from ℓ^1 -embeddability [DL94b], over the layout of electronic circuits [BGJR88, DSDJ⁺95], to solving Ising spin glass models, which are of high interest in physics [Bar82]; also see [DL94a, DL94b] for a more detailed overview on applications.

In contrast to MAXCUT the *minimum cut problem* and the *minimum s-t-cut problem*, MINCUT and s-t-MINCUT for short, are known to be solvable in polynomial time [FF56]. Formally, given a graph G , non-negative edge weights $c_e \in \mathbb{R}_{\geq 0}$, MINCUT asks for a non-empty cut $\delta_G(S)$ minimizing $\sum_{e \in \delta(S)} c_e$; s-t-MINCUT additionally receives two nodes $s, t \in V$ as input and asks for a minimum cut with $|S \cap \{s, t\}| = 1$. Research on these two problems is driven by applications including network design, network reliability, graph partitioning, partitioning items in databases, and information retrieval [PQ80, Bot93, Kar01].

A commonly used method to tackle graph optimization problems is linear programming. Here, we transfer the graph optimization to the respective optimization of a linear function over a polyhedron. One way to solve such problems is using branch-and-cut algorithms (see Chapter 1.3 for details). Using this approach, knowing facet-defining inequalities of the considered polyhedra as well as separation routines for these inequalities can massively speed up the computa-

tions. Hence, considering polyhedra arising from graph optimization problems, it is particularly interesting to find their *linear description*, i.e., their facet-defining inequalities. Moreover, the question arises, whether there are polynomial-time separation algorithms for these inequalities.

Approaching MAXCUT and s-t-MINCUT by linear programming gives rise to the cut polytope and the minimum s - t -cut dominant. The *cut polytope* $\text{CUT}^\square(G)$ is defined as the convex hull of all incidence vectors of cuts, the *minimum s - t -cut polytope* $\text{MINC}^\square(G, \{s, t\})$ is the convex hull of all incidence vectors of s - t -cuts, and the *minimum s - t -cut dominant* is given as $\text{MINC}(G, \{s, t\}) = \text{MINC}^\square(G, \{s, t\}) + \mathbb{R}^E$. MAXCUT can be solved by maximization over $\text{CUT}^\square(G)$; s-t-MINCUT can be solved by minimization over $\text{MINC}^\square(G, \{s, t\})$ or $\text{MINC}(G, \{s, t\})$. Since the vertices of the minimum cut dominant are precisely the incidence vectors of (inclusion-wise) minimal s - t -cuts, it is the relevant object for minimization. The minimum cut dominant was studied in [SW10]. Besides characterizing its vertices and adjacency among them, a complete facets-description of the minimum cut dominant is given.

Complementing the complexity gap between s-t-MINCUT and MAXCUT the cut polytope is the more complex of the two polyhedra. Although cut polytopes of complete graphs have been studied intensively (see, e.g., [DL92a, DL92b, DL10]), there are still many open problems; the facets of complete graphs on up to 7 nodes have been classified up to symmetry [DL10] and those of K_8 have been computed [CR01, DS16]. Determining the facets of $\text{CUT}^\square(K_n)$ for $n \geq 9$ remains an open problem.

Even less is known about cut polytopes of general graphs. There are a few graph classes for which complete descriptions of the cut polytope are known, e.g., K_5 -minor-free graphs [BM86]. Moreover, in [BM86] Barahona and Mahjoub provided an extensive study of the effect of graph operations such as node splittings, edge subdivisions, and edge contractions on the cut polytope and its facet-defining inequalities. Many questions regarding cut polytopes are still open, see e.g., Chapter 2.3.

A variety of cut problems arise from MAXCUT and s-t-MINCUT by generalization or by adding additional restrictions to the cuts under considerations. Over the time, many of those have been studied, e.g., k -cut problems [DGL91, DGL92] and equicut problems [dSL95, BB97, BCR97, DT98].

This thesis studied different cut problem on graphs. Mainly we focus our attention on the polyhedra arising from these problems by considering the respective linear programs. The original content is contained in Chapters 2 to 5. These chapters are based on the publications [CJNR22] (Chapter 2) and [CDJ⁺20] (Chapter 3), and the preprints [CJN20] (Chapter 4) and [CJN21] (Chapter 5). At the end of each chapter we collect open problems arising from the work on the respective contents.

Besides the work presented in this thesis, the author was also involved in research on properties of 2-crossing critical graphs [BCN⁺21]. Since crossing criticality is out of scope for this thesis, these results are not included.

Organization of the Thesis. In Chapter 1, we introduce basic results and terminology on graphs, polyhedra, and (integer) linear programming. Afterwards, we introduce the maximum cut problem. We provide an overview on complexity results for this problem as well as on previous work on cut polytopes which form the starting point for the following chapters.

In Chapter 2, we continue the research on cut polytopes. Complementing the results from [BM86] on K_5 -minor-free graphs, we give the facet-description of cut polytopes of $K_{3,3}$ -minor-free graphs and introduce an algorithm solving MAXCUT on those graphs in the same running time as planar MAXCUT. Moreover, starting a systematic geometric study of cut polytopes, we classify graphs admitting a simple or simplicial cut polytope.

In Chapter 3, we consider MAXCUT from a more in-depth algorithmic point of view. We propose a fixed-parameter tractable algorithm parameterized by the number k of crossings in a given drawing of G . Our algorithm achieves a running time of $\mathcal{O}(2^k \cdot p(n+k))$, where p is the polynomial running time for planar MAXCUT. The only previously known similar algorithm [DKM18] is restricted to graphs with at most one crossing per edge and its dependency on k is of the order of 3^k . Finally, combining this with the fact that the crossing number problem is fixed-parameter tractable with respect to its objective value, we see that MAXCUT is fixed-parameter tractable with respect to the crossing number, even without a given drawing. Moreover, the results naturally carry over to the minor-monotone-version of the crossing number.

In Chapter 4, we consider the maximum bond problem, MAXBOND for short—a variant of MAXCUT. There, given a graph $G = (V, E)$, we ask for a maximum bond, i.e. a maximum cut $\delta(S) \subseteq E$ with $S \subseteq V$ under the restriction that both $G[S]$ as well as $G[V \setminus S]$ are connected. We emphasize that connectivity arises naturally in optimal solutions of s-t-MINCUT. Thus, both MAXBOND and MAXCUT can be seen as inverse problems to s-t-MINCUT. While MAXCUT and its corresponding polytope have received a lot of attention in literature, comparably little is known about maximum bond.

The bond polytope is the convex hull of all incidence vectors of bonds. Similar to the connection of the corresponding optimization problems, the bond polytope is closely related to the cut polytope. In contrast to the intensive study of cut polytopes, previous to our research were no results on bond polytopes. We start a structural study of the latter, which additionally allows us to deduce algorithmic consequences.

We investigate the relation between cut- and bond polytopes and the additional intricacies that arise when requiring connectivity in the solutions. We study the effect of graph modifications on bond polytopes and their facets; moreover, we study facet-defining inequalities supported on edges and cycles. In particular, these yield a complete linear description of bond polytopes of cycles and 3-connected planar $(K_5 - e)$ -minor-free graphs. Finally, we present a reduction of the maximum bond problem on arbitrary graphs to the maximum bond problem on 3-connected graphs. This yields a linear-time algorithm for maximum bond on $(K_5 - e)$ -minor-free graphs.

In Chapter 5, we consider the minimum multicut problem, `MINMULTICUT` for short—a generalization of the minimum cut problem. Given a graph $G = (V, E)$ and a set $S \subseteq \binom{V}{2}$ of terminal pairs, the minimum multicut problem asks for a minimum edge set $\delta \subseteq E$ such that there is no s - t -path in $G - \delta$ for any $\{s, t\} \in S$. For $|S| = 1$, this is the minimum s - t -cut problem, but in general the minimum multicut problem is NP-complete, even if the input graph is a tree. The multicut polytope $\text{MULTC}^\square(G, S)$ is the convex hull of all incidence vectors of multicuts in G ; the multicut dominant is given by $\text{MULTC}(G, S) = \text{MULTC}^\square(G, S) + \mathbb{R}^E$. The latter is the relevant object for the minimization problem. While polyhedra associated to several other cut problems have been studied intensively there is only little knowledge for multicut.

We investigate properties of the multicut dominant and in particular derive results on liftings of facet-defining inequalities. This yields a classification of all facet-defining path- and edge inequalities. Moreover, we investigate the effect of graph operations on the multicut-dominant and its facet-defining inequalities. In addition, we introduce facet-defining inequalities supported on stars, trees, and cycles and show that the former two can be separated in polynomial time when the input graph is a tree.

Chapter 1

Preliminaries

In this chapter we provide some background used in the thesis. After giving basic results and terminology for graphs and polyhedra, we briefly discuss integer linear programming. Then, we introduce the maximum cut problem and close the chapter by introducing the cut polytope and recapitulating previous work on it.

1.1 Graphs

In this section, we provide basic background on graphs. For more details, we refer to [Die18].

An (undirected) *simple graph* is a pair $G = (V, E)$ where V is a set and $E \subseteq \binom{V}{2}$ is a set of two-element subsets of V . Elements of V are called *nodes* or *vertices*; elements of E are called edges. Given a graph G , we also write $V(G)$ and $E(G)$ for its set of nodes and its set of edges, respectively. For $v, w \in V(G)$, let $vw = \{v, w\}$ be the edge between v and w . We call v, w the *end nodes* of vw . Two nodes v and w are *adjacent* if $vw \in E(G)$; two edges $e, f \in E(G)$ are *adjacent* if $e \cap f \neq \emptyset$. An edge $e \in E(G)$ and a node $v \in V(G)$ are *incident* if $v \in e$.

Multigraphs generalize simple graphs by allowing multiple edges with the same end nodes. Unless specified otherwise, we only consider simple undirected graphs that contain no isolated nodes.

A *path* of length k is a sequence of edges e_1, \dots, e_k with $e_i = v_{i-1}v_i$ such that $v_i \neq v_j$ for $0 \leq i < j \leq k$. Such a sequence but with $v_0 = v_k$ is a *cycle*; a cycle of length 3 is a *triangle*. A graph H is a *subgraph* of G , denoted by $H \subseteq G$, if (after possibly renaming) $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. Given a subset $W \subseteq V$, the subgraph *induced* by W is the graph $G[W] = (W, \{uv \in E : u, v \in W\})$. If an induced subgraph forms a cycle, this is an *induced cycle* and thus *chordless*. A graph G is *chordal*, if every induced cycle in G has length 3. We fix the following notations for some special classes of graphs: C_n for the cycle of length n ; K_n for the complete graph on n nodes; $K_{n,m}$ for the complete bipartite graph on n and m

nodes per partition set. We denote the graph obtained from G by deleting nodes v_1, \dots, v_k (resp. edges e_1, \dots, e_k) by $G - \{v_1, \dots, v_k\}$ (resp. $G - \{e_1, \dots, e_k\}$). If we remove a single node v (resp. edge e), we might just write $G - v$ (resp. $G - e$).

The graph G/e is obtained from G by *contracting* edge $e = vw$, i.e., the nodes v and w are identified, and we delete the arising self-loop and merge parallel edges. A graph G contains an *H-minor*, if H can be obtained from G by contracting and deleting edges and isolated nodes; otherwise G is *H-minor-free*; G is an *H-subdivision*, if G is obtained from H by replacing edges by internally node-disjoint paths.

G is *k-connected* if $|V(G)| \geq k+1$ and for each pair of nodes $v, w \in V(G)$ there exist k (internally) node-disjoint paths from v to w . 1-connected graphs as well as the graph consisting of a single node and the empty graph are called *connected*. If G is 1-connected but not 2-connected, there exists some *cut-node* $v \in V(G)$ such that $G - v$ is disconnected.

For two graphs G and H , their *union* $G \cup H = (V(G) \cup V(H), E(G) \cup E(H))$ is disjoint if their node sets are; in this case we write $G \uplus H$. Consider two graphs G, H containing K_k as a subgraph, for some $k \in \mathbb{N}_{>0}$, the *k-sum* (or *clique-sum*) of G and H is obtained by taking their union, identifying the K_k subgraphs and possibly removing edges contained in this specific K_k . A *k-sum* is *strict*, if no edges are removed. We denote the strict *k-sum* of G and H by $G \oplus_k H$. Observe, that this notation does not explicitly state the specific K_k in question.

A *drawing* of a graph (in the plane) consists of a map of its nodes to distinct points in \mathbb{R}^2 together with a map of its edges to curves connecting the respective endpoints. The interior of the curve of an edge must not contain the point of any node. Any point in the plane either corresponds to a graph node, or is contained in at most two edge curves. A shared non-endpoint between two curves is called a *crossing*. A graph is *planar* if it admits a drawing without any crossings. (Edge-)maximal planar graphs are *triangulations*. By Kuratowski's (Wagner's) Theorem [Kur30, Wag37], a graph is planar if and only if it contains no K_5 - or $K_{3,3}$ -subdivision (minor, respectively). Given a K_5 -subdivision H contained in G we call the nodes of degree 4 in H *Kuratowski nodes*. The paths in H between these nodes are *Kuratowski paths*.

1.2 Polyhedra

In this section we provide basic background on polyhedra. For more details, we refer to [BG09, Zie12].

For $k \in \mathbb{N}$, let $[k] = \{1, \dots, k\}$; for vectors a, b in \mathbb{R}^d , we write $a \leq b$ if $a_i \leq b_i$ for all $i \in [d]$; for $A \in \mathbb{R}^{n \times d}$ and $b \in \mathbb{R}^d$ we use the shorthand $\{Ax \leq b\}$ for $\{x \in \mathbb{R}^d : Ax \leq b\}$ and the analogon for equalities. Given a halfspace

$H = \{a^\top x \leq b_0\}$ with $a \in \mathbb{R}^d$ and $b_0 \in \mathbb{R}$ we call the vectors in $\{\lambda a : \lambda \in \mathbb{R}_{>0}\}$ *outer normals* of H and those in $\{-\lambda a : \lambda \in \mathbb{R}_{>0}\}$ *inner normals*. While we describe halfspaces by their outer normals in the following discussion, everything can be done similarly using inner normals as $\{a^\top x \leq b_0\} = \{-a^\top x \geq -b_0\}$. As a rule of thumb we will use outer normals (\leq -inequalities) when maximizing some objective function over a polyhedron and inner normals (\geq -inequalities) when minimization is considered.

A \mathcal{V} -polytope is the convex hull of a finite set of points in \mathbb{R}^d , i.e., it is a set of the form $\mathcal{P} = \text{conv}(M)$ for some finite set $M \subseteq \mathbb{R}^d$. The *Minkowski sum* of two sets $\mathcal{P}, \mathcal{Q} \subseteq \mathbb{R}^d$ is given by $\mathcal{P} + \mathcal{Q} = \{p + q : p \in \mathcal{P}, q \in \mathcal{Q}\}$. A \mathcal{V} -polyhedron is the Minkowski sum of a \mathcal{V} -polytope and the conical hull of a finite set, i.e., it is a set of the form $\mathcal{P} = \text{conv}(M) + \text{cone}(N)$ for finite sets $M, N \subseteq \mathbb{R}^d$. A \mathcal{V} -polyhedron is a \mathcal{V} -polytope if and only if it is bounded.

An \mathcal{H} -polyhedron is the intersection of finitely many closed halfspaces, i.e., it is a set of the form $\mathcal{P} = \{Ax \leq b\} \subseteq \mathbb{R}^d$ for some $A \in \mathbb{R}^{m \times d}$ and $b \in \mathbb{R}^m$. An \mathcal{H} -polytope is a bounded \mathcal{H} -polyhedron. Due to the following theorem by Minkowski and Weyl, we will not explicitly distinguish between \mathcal{H} - and \mathcal{V} -polyhedra and just call these objects polyhedra.

Theorem 1.2.1. *Let $\mathcal{P} \subseteq \mathbb{R}^d$. Then, \mathcal{P} is an \mathcal{H} -polyhedron (resp. \mathcal{H} -polytope) if and only if \mathcal{P} is an \mathcal{V} -polyhedron (resp. \mathcal{V} -polytope).*

In the following, let \mathcal{P} be a polyhedron. The *dimension* $\dim \mathcal{P}$ of \mathcal{P} is the dimension of its affine hull. A linear inequality $a^\top x \leq b_0$ with $a \in \mathbb{R}^d$ and $b_0 \in \mathbb{R}$ is a *valid inequality* for \mathcal{P} if it is satisfied by all points $x \in \mathcal{P}$. The inequality is *tight* if there is some $p \in \mathcal{P}$ with $a^\top p = b_0$. A (proper) *face* of \mathcal{P} is a (non-empty) set \mathcal{F} of the form $\mathcal{F} = \mathcal{P} \cap \{a^\top x = b_0\}$ for some valid inequality $a^\top x \leq b_0$ with $a \neq \mathbf{0}$. Faces of dimension 0 and $\dim(\mathcal{P}) - 1$ are *vertices* and *facets*, respectively. For each face $\mathcal{F} \subseteq \mathcal{P}$ there is a facet $\mathcal{F}' \subseteq \mathcal{P}$ *dominating* it, i.e., $\mathcal{F}' \supseteq \mathcal{F}$.

A tight inequality $a^\top x \leq b_0$ is *facet-defining* if $\mathcal{P} \cap \{a^\top x = b_0\}$ is a facet of \mathcal{P} . If $\mathcal{P} = \{Ax \leq b\}$ for some matrix A and vector b , the system of inequalities $Ax \leq b$ is a *linear description* of \mathcal{P} . If \mathcal{P} is full-dimensional a minimal linear description is given by taking the system of all facet-defining inequalities.

A *simplex* of dimension d is the convex hull of $d+1$ affinely independent points. A d -dimensional polytope \mathcal{P} is *simple* if each vertex of \mathcal{P} is contained in exactly d facets; the polytope is *simplicial* if each facet of \mathcal{P} is a simplex.

Given Theorem 1.2.1, it is straightforward to verify that both the intersection of a polyhedron with a hyperplane and the projection of a polyhedron onto a linear subspace are again polyhedra. Moreover, given the linear description of a polyhedron \mathcal{P} we can derive a linear description of the orthogonal projection of \mathcal{P} onto a coordinate hyperplane using Fourier-Motzkin elimination. An example for this can be found in Example 2.1.5.

Theorem 1.2.2 (Fourier-Motzkin elimination). *Let $\mathcal{P} = \{Ax \leq b\}$ be a polyhedron with $A \in \mathbb{R}^{m \times d}$ and $b \in \mathbb{R}^m$ and $k \in [d]$. Denote by $\pi_k: \mathbb{R}^d \rightarrow \mathbb{R}^{d-1}$ the orthogonal projection onto $\{x_k = 0\}$. Let $A_i = (a_{i,1}, \dots, a_{i,k-1}, a_{i,k+1}, \dots, a_{i,d})$ be obtained from the i -th row of A by ignoring the k -th entry.*

Then, the following system of inequalities is a linear description of $\pi(\mathcal{P})$:

$$\begin{aligned} A_i x &\leq b_i && \text{for all } i \in [d] \text{ with } a_{i,k} = 0, \\ (a_{i,k} A_j - a_{j,k} A_i) x &\leq a_{i,k} b_j - a_{j,k} b_i && \text{for all } i, j \in [d] \text{ with } a_{i,k} > 0 \text{ and } a_{j,k} < 0. \end{aligned}$$

1.3 Integer Linear Programming

A common way to approach optimization problems is integer linear programming. In this section we provide some background on the fundamental ideas of this technique. For more details we refer to [NW88].

A *linear program* (LP) is the maximization or minimization of a linear function over a polyhedron, i.e., we ask for for

$$\max\{c^\top x : x \in \mathcal{P}\} \quad \text{or} \quad \min\{c^\top x : x \in \mathcal{P}\}$$

where $\mathcal{P} \subseteq \mathbb{R}^d$ is a polyhedron and $c \in \mathbb{R}^d$. The function $f(x) = c^\top x$ is called the *objective function*. A linear program is *infeasible* if there is no solution, i.e., if $\mathcal{P} = \emptyset$; otherwise it is *feasible*.

Given a polyhedron $\mathcal{P} \subseteq \mathbb{R}^d$, its *integer hull* is $\mathcal{P}_I = \text{conv}\{x \in \mathcal{P} : x \in \mathbb{Z}^d\}$. We also say \mathcal{P} is a *relaxation* of \mathcal{P}_I . An *integer linear program* (ILP) is the maximization or minimization of a linear function over the integer points in a polyhedron, i.e., the question for

$$\max\{c^\top x : x \in \mathcal{P} \cap \mathbb{Z}^d\} \quad \text{or} \quad \min\{c^\top x : x \in \mathcal{P} \cap \mathbb{Z}^d\},$$

where $\mathcal{P} \subseteq \mathbb{R}^d$ is a polyhedron and $c \in \mathbb{R}^d$. In terms of the considered vertices and the optimal value, this is equivalent to the respective linear program over \mathcal{P}_I . However, while linear programs can be solved in polynomial time, solving integer linear programs is in general NP-complete.

One way to solve an integer linear program is to use the *branch-and-cut* method. In the following we describe this procedure for maximization; this method can be transferred to minimization by simply considering the negative of the objective function. To this end, let \mathcal{P} be a polyhedron and consider the integer linear program $\max\{c^\top x : x \in \mathcal{P} \cap \mathbb{Z}^d\}$.

A *cutting plane* is a hyperplane $H = \{a^\top x = b\}$ such that the corresponding *cutting plane inequality* $a^\top x \leq b$ is valid for \mathcal{P}_I but not for \mathcal{P} . Given a class of cutting planes and some $x \in \mathbb{R}^d$, an (exact) *separation algorithm* gives a violated inequality of this class if it exists.

Branch-and-cut solves the integer linear program by an enumeration starting with the relaxation \mathcal{P} . Formally, we perform the following steps:

0. *Initialization:* Set $\Pi = \{\mathcal{P}\}$, $x^* = \mathbf{0}$ and $v^* = -\infty$.
1. *Termination:* If $\Pi = \emptyset$, terminate with optimal solution x^* of value v^* .
2. *LP Selection:* Select a polyhedron $\mathcal{Q} \in \Pi$ and remove it from Π .
3. *LP Solving:* Solve the linear program $\max\{c^\top x : x \in \mathcal{Q}\}$.
4. *Bounding:* If the linear program is infeasible or $v \leq v^*$, go back to step 2.
5. *Update Bound:* If $\hat{x} \in \mathbb{Z}^d$, set $v^* = v$, $x^* = \hat{x}$, and go back to step 2.
6. *Cutting:* We may check whether there are cutting planes separating \hat{x} from \mathcal{Q}_I . If such a cutting plane inequality $a^\top x \leq b$ is found, update $\mathcal{Q} = \mathcal{Q} \cap \{a^\top x \leq b\}$ and go back to step 3.
7. *Branching:* Choose some index $i \in [d]$ with $\hat{x}_i \notin \mathbb{Z}$, add $\mathcal{Q} \cap \{x_i \geq \lceil \hat{x}_i \rceil\}$ and $\mathcal{Q} \cap \{x_i \leq \lfloor \hat{x}_i \rfloor\}$ to Π , and go back to step 2.

In each step we have a collection Π of *active* polyhedra and apply the following steps to a chosen polyhedron. We start with *bounding*, i.e., we check whether the maximization over the chosen polyhedron yields a larger value than the best feasible solution value seen so far. If it does not, we remove the chosen polyhedron from the list of active polyhedra; otherwise, if it does and the found point attaining this value is integral, we update the best feasible solution value and discard the polyhedron as well. If neither of these cases applies, we try to *cut* the polyhedron with a halfspace defined by some cutting plane and resolve the corresponding linear program. If there is no identified cutting plane, we *branch*, i.e., we generate linear programs as subproblems which are then added to the collection of active polyhedra instead of the chosen polyhedron. Note that in each iteration cutting is optional and dependent on the set of known cutting planes and separation routines for these. The procedure terminates when the list of active polyhedra is empty. The at this point best found solution is in fact optimal.

Note that while branching always adds two new active polyhedra, the number of active polyhedra does not grow as long as cutting is successful. Thus, the procedure performs better when “good” cutting planes—i.e., cutting planes yielding inequalities that are “as tight as possible”—can be found. The best candidates for these hyperplanes are the facets of \mathcal{P}_I that are not facets of \mathcal{P} . Thus, determining facets of \mathcal{P}_I and finding separation algorithms for these hyperplanes can substantially speed up the branch-and-cut procedure.

1.4 The Maximum Cut Problem

Let $G = (V, E)$ be a graph. A selection $S \subseteq V$ of nodes defines the *cut* $\delta_G(S) = \{e \in E : |e \cap S| = 1\}$. We may omit the subscript when the graph is clear from the context. If G is connected, there are $2^{|V|-1}$ pairwise different cuts as $\delta(S) = \delta(V \setminus S)$. Given a graph G and edge weights $c_e \in \mathbb{R}$, for $e \in E$, the maximum cut problem, MAXCUT problem for short, asks for a cut δ in G that maximizes its *value* $\sum_{e \in \delta} c_e$. Since a graph can have multiple cuts of equal value, only the value of a maximum cut is unique, not the cut itself.

It is well-known that MAXCUT is NP-complete [Kar72]. Papadimitriou and Yannakakis [PY91] showed that it is even APX-hard, i.e., there does not exist a polynomial-time approximation scheme unless $P=NP$. A few years later, Goemans and Williamson proposed a randomized constant-factor approximation algorithm [GW95], which has been derandomized by Mahajan and Ramesh [MR95], achieving a ratio of 0.87856. Although MAXCUT is NP-complete on general graphs, there are some graph classes on which polynomial-time algorithms are known. In [OD72, Had75] it was shown that MAXCUT can be solved in polynomial time for unweighted planar graphs. This result can be extended to the weighted case; the currently fastest algorithms for the weighted case have been suggested by Shih et al. [SWK90] and by Liers and Pardella [LP12], and achieve a running time of $p(n) = \mathcal{O}(n^{3/2} \log n)$ on planar graphs with n nodes.

Extending the class of planar graphs, Barahona [Bar83] proposed a polynomial-time algorithm solving MAXCUT on K_5 -minor-free graphs in $\mathcal{O}(n^4)$ time. This was generalized by Kaminski [Kam12] by proving that MAXCUT can be solved in $\mathcal{O}(n^4)$ time on H -minor-free graphs, for an arbitrary graph H that admits a drawing with exactly one crossing. A further extension of the class of K_5 -minor-free graphs was given by Grötschel and Pulleyblank by introducing *weakly bipartite* graphs [GP81, FMU92]. These are by definition the graphs, whose bipartite subgraph polytope is completely described by certain edge- and cycle inequalities (see Theorem 1.5.3). Moreover, they proved that for positive edge weights, MAXCUT can be solved in polynomial time on these graphs by using linear programming. In contrast to these results, MAXCUT is still NP-complete on K_6 -minor-free graphs [Bar83].

1.5 Cut Polytopes

Given a graph $G = (V, E)$, we associate to each set $\delta \subseteq E$ its *incidence vector* $x^\delta \in \mathbb{R}^E$ given by

$$x_e^\delta = \begin{cases} 1, & \text{if } e \in \delta, \\ 0, & \text{else.} \end{cases}$$

The *cut polytope* of G is defined as the convex hull of the incidence vectors of all cuts in G :

$$\text{CUT}^\square(G) = \text{conv}(\{x^\delta : \delta \text{ is a cut in } G\}) \subseteq \mathbb{R}^E.$$

The cut polytope has dimension $\dim(\text{CUT}^\square(G)) = |E|$, see, e.g., [BGM85, p.344]. Given a valid inequality $a^\top x \leq b$ of $\text{CUT}^\square(G)$ with $a \in \mathbb{R}^E$ and $b \in \mathbb{R}$, its *support graph* $\text{supp}(a) \subseteq G$ is the subgraph of G induced by the edge set $\{e \in E : a_e \neq 0\}$. If $\text{supp}(a)$ consists of a single edge, the inequality is called an *edge inequality*. The *homogeneous edge inequality* associated to an edge $e \in E$ is $-x_e \leq 0$.

In the following we recapitulate results on cut polytopes of general graphs.

Graph Decompositions. It follows directly from the definition of cut polytopes that for disconnected graphs G with connected components $G_1 \dots G_k$ we have

$$\text{CUT}^\square(G) = \text{CUT}^\square(G_1) \times \dots \times \text{CUT}^\square(G_k). \quad (1.1)$$

As many classes of graphs can be described in terms of clique sums, the following result is a very helpful cornerstone for the understanding of cut polytopes.

Theorem 1.5.1. [Bar83, Theorem 3.1.] *Let $G = G_1 \oplus_k G_2$ be a strict k -sum with $k \in \{1, 2, 3\}$. The facet-defining inequalities of G are given by taking all facet-defining inequalities of G_1 and G_2 and identifying the variables of common edges. In particular, it holds that*

$$\text{CUT}^\square(G_1 \oplus_1 G_2) = \text{CUT}^\square(G_1) \times \text{CUT}^\square(G_2). \quad (1.2)$$

Symmetry Operations. Any automorphism ϕ of a graph G gives rise to a map on cuts. Thus, ϕ induces a permutation on the vertices of $\text{CUT}^\square(G)$ by mapping x^δ to $x^{\phi(\delta)}$, which yields a symmetry of $\text{CUT}^\square(G)$. Another symmetry of cut polytopes is given by *switching*:

Lemma 1.5.2 (Switching Lemma [BM86, Corollary 2.9.]). *Let $G = (V, E)$ be a graph and $a^\top x \leq b$ be a facet-defining inequality for $\text{CUT}^\square(G)$. Let $W \subseteq V$, and define $b' = b - \sum_{e \in \delta(W)} a_e$, and $a'_e = (-1)^{\mathbf{1}[e \in \delta(W)]} \cdot a_e$ for all $e \in E$. Then $(a')^\top x \leq b'$ defines a facet of $\text{CUT}^\square(G)$.*

Considering this on the level of cuts, switching in $\delta(W)$ is induced by the map $\delta \mapsto \delta \Delta \delta(W) = (\delta \cup \delta(W)) \setminus (\delta \cap \delta(W))$. In particular, using notations of the previous lemma, this maps the vertices on a facet $\text{CUT}^\square(G) \cap \{a^\top x = b\}$ onto the vertices of the facet $\text{CUT}^\square(G) \cap \{(a')^\top x = b'\}$. Moreover, switching a facet-defining inequality by a cut corresponding to a vertex of this facet gives a homogeneous facet-defining inequality. Thus, all symmetry classes of facets of $\text{CUT}^\square(G)$ contain facets of the *cut cone* $\text{CUT}(G) = \text{cone}(\{x \in \text{CUT}^\square(G)\}) \subseteq \mathbb{R}^E$. It hence suffices to understand the facets of cut cones to understand the facets of cut polytopes.

Facet-Defining Inequalities. Considering cut polytopes, it is particularly interesting to find their *linear description*, i.e., their facet-defining inequalities. If we have a linear description of polynomial size in the input, this gives a polynomial algorithm for MAXCUT. Even though it is impossible to find such a description for arbitrary graphs (unless $P=NP$), a better understanding of cut polytopes is expected to improve algorithmic results.

Since $\text{CUT}^\square(G)$ is contained in the unit cube, the inequalities $0 \leq x_e \leq 1$ are valid. Given a cut δ and a cycle C in G , the number of edges in $\delta \cap C$ is always even. These observations give rise to validity of the following *edge-* and *cycle inequalities*:

Theorem 1.5.3. [BM86, Section 3]

- The valid inequalities $0 \leq x_e \leq 1$ define facets of $\text{CUT}^\square(G)$ if and only if e does not belong to a triangle.
- The valid inequalities

$$\sum_{f \in F} x_f - \sum_{e \in E(C) \setminus F} x_e \leq |F| - 1, \text{ for all cycles } C \subseteq G, F \subseteq E(C) \text{ with } |F| \text{ odd}$$

define facets if and only if C is chordless.

In particular, for each triangle G with $E(G) = \{e, f, g\}$ the following *metric inequalities* (up to permuting the edges) are facet-defining for $\text{CUT}^\square(G)$:

$$x_e + x_f + x_g \leq 2 \quad \text{and} \quad x_e - x_f - x_g \leq 0.$$

As a generalization of metric inequalities we get *hypermetric* inequalities by considering the complete graph instead of triangles. Up to switching these inequalities are given by the following theorem:

Theorem 1.5.4. [BM86, Theorem 2.2 and 2.4] Let $G = (V, E)$ be a graph, $n \geq 5$ and $H = ([n], F) \subseteq G$ be a copy of K_n . Moreover, let $k \geq 2$ and $t_1, \dots, t_n \in \mathbb{N}_{>0}$ with $\sum_{i=1}^n t_i = 2k + 1$ and $\sum_{i \in [n]: t_i > 1} t_i \leq k - 1$. Then,

$$\sum_{ij \in F} t_i t_j x_{ij} \leq k(k + 1)$$

is facet-defining for $\text{CUT}^\square(G)$.

The hypermetric inequality of K_5 is displayed in inequality (2.1). The support of all facet-defining inequalities treated so far correspond to complete subgraphs

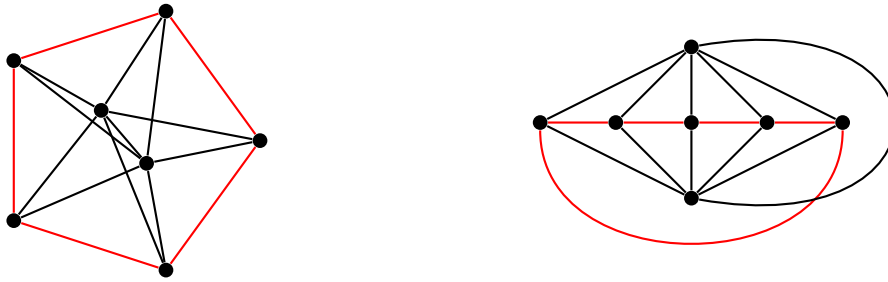


Figure 1.1: Two drawings of the bicycle 5-wheel. The red edges are those of the cycle C_n .

or, in the case of cycle inequalities, subdivisions of these. There also exist facet-defining inequalities whose support graph is not complete. One example for this type of inequalities is given by *bicycle n -wheel inequalities*: A *bicycle n -wheel*, cf. Figure 1.1, is the graph obtained from the cycle C_n by adding two adjacent nodes v_1, v_2 and edges $v_i w$ for each $i \in [2]$ and $w \in V(C_n)$.

Theorem 1.5.5. [BM86, Theorem 2.3] *Let $G = (V, E)$ be a graph and $H = (W, F)$ be a bicycle $(2k + 1)$ -wheel contained in G . Then, the inequality*

$$\sum_{f \in F} x_f \leq 2(2k + 1)$$

is facet-defining for $\text{CUT}^\square(G)$.

For complete graphs, further classes of facet-defining inequalities of the cut polytope are given in [AI07, DL10]. In particular, for $n \leq 7$ all facets of $\text{CUT}^\square(K_n)$ are classified up to symmetry [DL10, Chapter 30.6] and all facets of $\text{CUT}^\square(K_8)$ have been computed (see [CR01, Section 8.3] and [DS16]). It is a major open problem to determine the facets of $\text{CUT}^\square(K_n)$ for $n \geq 9$.

A main result in the study of cut polytopes of general graphs is the classification of K_5 -minor-free graphs by the facet-defining inequalities of their cut polytopes:

Theorem 1.5.6. [BM86, Section 3] *A graph G is K_5 -minor-free if and only if $\text{CUT}^\square(G)$ is completely defined by the cycle- and edge inequalities from Theorem 1.5.3.*

Graph Operations. We close this section by presenting results on the effect of graph operations on the cut polytope and its facet-defining inequalities. As a first step, note that for a graph $G = (V, E)$ and an edge $e \in E$, cuts in G/e correspond to cuts δ in G with $e \notin \delta$. Moreover, if δ is a cut in G , then $\delta \setminus \{e\}$ is a cut in $G - e$ and each cut in $G - e$ can be constructed this way. Thus, the following observation holds:

Observation 1.5.7. *Let $G = (V, E)$ be a graph and $e \in E$. Then,*

- $\text{CUT}^\square(G/e) = \text{CUT}^\square(G) \cap \{x_e = 0\}$, and
- $\text{CUT}^\square(G - e) = \pi(\text{CUT}^\square(G))$, where $\pi: \mathbb{R}^E \rightarrow \mathbb{R}^{E \setminus \{e\}}$ is the orthogonal projection.

Finally, we recapitulate more in-depth results on the effect of node splittings, edge subdivisions, and edge contractions on facet-defining inequalities of cut polytopes. Though stating the following theorems is quite tedious and the precise results are not used directly in the following work, we present these for easier comparison with the respective results in Chapter 4.2 and Chapter 5.2 We start by considering a simple graph operation:

Theorem 1.5.8 (Contraction of an edge [BM86, Theorem 2.6(b)]). *Let $G = (V, E)$ be a graph, $a^\top x \leq b$ be facet-defining for $\text{CUT}^\square(G)$, and $v_1 v_2 \in E(\text{supp}(a))$ such that v_1 and v_2 have no common neighbor in $\text{supp}(a)$. Assume that $a_{v_1 u} \geq 0$ for each $v_1 u \in \delta(v_1) \setminus \{v_1 v_2\}$ and $-a_{v_1 v_2} = \sum_{e \in \delta(v_1) \setminus \{v_1 v_2\}} a_e \geq \sum_{e \in \delta(v_2) \setminus \{v_1 v_2\}} a_e$. Obtain \bar{G} from G by removing all edges $v_1 u$ with $a_{v_1 u} = 0$ and then contracting the edge $v_1 v_2$. Define $\bar{a} \in \mathbb{R}^{\bar{E}}$ by*

$$\bar{a}_{uw} = \begin{cases} a_{uw}, & \text{if } uw \in E \cap \bar{E}, \\ a_{v_1 w} & \text{if } u = v_1 \text{ and } a_{v_1 w} > 0, \\ a_{v_2 w} & \text{if } u = v_2, v_2 w \in E, \text{ and } v_1 w \notin E(\text{supp}(a)). \end{cases}$$

Then, $\bar{a}^\top x \leq b$ is facet-defining for $\text{CUT}^\square(\bar{G})$.

After considering edge contractions, we consider its inverse operation:

Theorem 1.5.9 (Node splitting [BM86, Theorem 2.6(a)]). *Let $G = (V, E)$ be a graph, $a^\top x \leq b$ be a facet-defining inequality for $\text{CUT}^\square(G)$ and $v \in V(\text{supp}(a))$. Furthermore, let $W \subseteq V$ such that $v \in W$ and $a^\top x^{\delta(W)} = b$. Choose any nonempty subset $F \subseteq \delta(v) \cap \{xy \in E : x, y \in W\}$ such that $a_e > 0$ for each $e \in F$ and construct the graph $\bar{G} = (\bar{V}, \bar{E})$ from G as follows: Split v into adjacent nodes v_1, v_2 such that v_1 is incident to each edge in F and v_2 is incident to each edge in $\delta(v) \setminus F$. Moreover, any further edge $v_1 u$ with $u \notin V(\text{supp}(a))$ may be added. Now, define $\bar{a} \in \mathbb{R}^{\bar{E}}$ by*

$$\bar{a}_{uw} = \begin{cases} a_{uw}, & \text{if } uw \notin \delta(v), \\ a_{uw}, & \text{if } u = v_1, uw \in F, \\ a_{uw}, & \text{if } u = v_2, uw \in (\delta(v) \cap E(\text{supp}(a))) \setminus F, \\ -\sum_{f \in F} a_f, & \text{if } uw = v_1 v_2, \\ 0, & \text{else.} \end{cases}$$

Then, $\bar{a}^\top x \leq b$ is facet-defining for $\text{CUT}^\square(\bar{G})$.

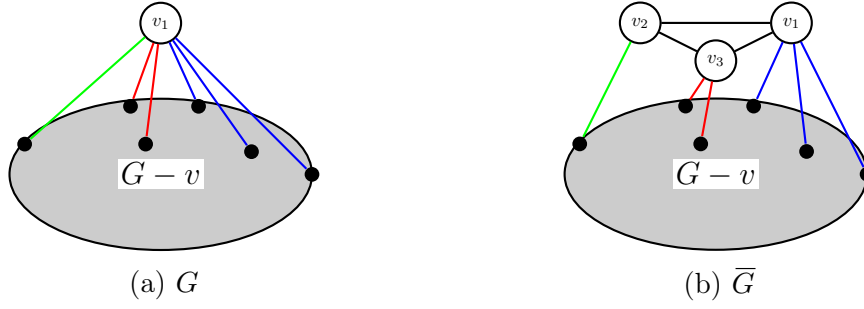


Figure 1.2: Visualization of the split operation in Theorem 1.5.10. Blue, green and red edges represent F_1 , F_2 , and F_3 , respectively.

Although node splitting can be used as inverse to edge contractions, we can also split a node into more than just two adjacent nodes:

Theorem 1.5.10 (Replacing a node by a triangle [BM86, Theorem 2.7]). *Let $G = (V, E)$ be a graph, $a^\top x \leq b$ be facet-defining for $\text{CUT}^\square(G)$, and $v \in V(\text{supp}(a))$ such that $a_e \geq 0$ for all $e \in \delta(v)$. Furthermore, let $F_1 \cup F_2 \cup F_3$ be a partition of $\delta(v)$. Assume that there exist $W_1, W_2, W_3 \subseteq V$ such that $a^\top x^{\delta(W_i)} = b$ and $F_i \subseteq \{vw \in W : v, w \in W_i\}$ for all $i \in [3]$. Construct $\bar{G} = (\bar{V}, \bar{E})$ from G as follows (cf. Figure 1.2): Remove v from G and add nodes v_1, v_2, v_3 such that v_i is incident to all edges in F_i for $i \in [3]$ and v_j is adjacent to v_k for $1 \leq j < k \leq 3$. Moreover, any edge $v_i u$ with $i \in [3]$ and $u \notin V(\text{supp}(a))$ may be added. Define $\bar{a} \in \mathbb{R}^{\bar{E}}$ by*

$$\bar{a}_{uw} = \begin{cases} a_{uw}, & \text{if } uw \in E \setminus \delta(v), \\ a_{uw}, & \text{if } u = v_i, i \in [3], uw \in F_i, \\ \frac{1}{2} \left(-\sum_{f \in F_1} a_f - \sum_{f \in F_2} a_f + \sum_{f \in F_3} a_f \right), & \text{if } uw = v_1 v_2, \\ \frac{1}{2} \left(-\sum_{f \in F_1} a_f + \sum_{f \in F_2} a_f - \sum_{f \in F_3} a_f \right), & \text{if } uw = v_1 v_3, \\ \frac{1}{2} \left(\sum_{f \in F_1} a_f - \sum_{f \in F_2} a_f - \sum_{f \in F_3} a_f \right), & \text{if } uw = v_2 v_3, \\ 0, & \text{else.} \end{cases}$$

Then, $\bar{a}^\top x \leq b$ is facet-defining for $\text{CUT}^\square(\bar{G})$.

After considering edge contractions and two different node split operations, we turn our attention to another simple graph operation:

Theorem 1.5.11 (Subdivision of an edge [BM86, Corollary 2.10(a)]). *Let $G = (V, E)$ be a graph, $a^\top x \leq b$ be facet-defining for $\text{CUT}^\square(G)$ and $vw \in E(\text{supp}(a))$. Let $\bar{G} = (\bar{V}, \bar{E})$ be the graph obtained from $G - vw$ in the following way: Add nodes*

v_1, \dots, v_k and edges $P = \{vv_1, v_1v_2, \dots, v_{k-1}v_k, v_kv\}$. Moreover, any further edge $v_\ell u$ with $1 \leq \ell \leq k$ and $u \notin V(\text{supp}(a))$ may be added. Let $P^+ \cup P^-$ be a partition of P such that $|P^+|$ is odd if $a_{vw} > 0$ and $|P^-|$ is even if $a_{vw} < 0$. Define $\bar{a} \in \mathbb{R}^{\bar{E}}$ by

$$\bar{a}_e = \begin{cases} a_e, & \text{if } e \in E \setminus \{vw\}, \\ a_{vw}, & \text{if } e \in P^+, \\ -a_{vw}, & \text{if } e \in P^-, \\ 0, & \text{else.} \end{cases}$$

Then, $\bar{a}^\top x \leq b$ is facet-defining for $\text{CUT}^\square(\bar{G})$.

Finally, after considering edge subdivisions, we consider its inverse operation:

Theorem 1.5.12 (Replacing a path by an edge [BM86, Corollary 2.10(b)]). *Let $G = (V, E)$ be a graph and $a^\top x \leq b$ be facet-defining for $\text{CUT}^\square(G)$. Moreover, let $P = \{vv_1, v_1v_2, \dots, v_{k-1}v_k, v_kv\}$ be an induced path in $\text{supp}(a)$. Assume there is some $\alpha \neq 0$ and a partition $P = P^+ \cup P^-$ with $a_e = \alpha$ for $e \in P^+$ and $a_e = -\alpha$ for $e \in P^-$ with $|P^+|$ is odd if $\alpha > 0$ and $|P^-|$ is odd if $\alpha < 0$. Let \bar{G} be obtained from G by removing P and adding the edge vw . Define $\bar{a} \in \mathbb{R}^{\bar{E}}$ by*

$$\bar{a}_e = \begin{cases} a_e, & \text{if } e \in E \cap \bar{E}, \\ \alpha, & \text{else.} \end{cases}$$

Then, $\bar{a}^\top x \leq b$ is facet-defining for $\text{CUT}^\square(\bar{G})$.

Chapter 2

Cut Polytopes of Minor-free Graphs

This chapter is based on [CJNR22].

Not too long ago, Sturmfels and Sullivant [SS08] established a new connection between the study of cut polytopes and commutative algebra, as well as algebraic geometry, by considering related toric varieties. In particular, they conjectured that the cut polytope of a graph is normal if and only if the graph is K_5 -minor-free. Among others, the research on these toric varieties and associated cut algebras has been pursued by Engström [Eng11], Ohsugi [Ohs10, Ohs14], Römer and Saeedi Madani [RS18], Römer and Koley [KR21], and Lasoń and Michałek [LM20].

It turns out that not much is known about the polyhedral structure of cut polytopes as objects in discrete geometry. We expect new insights in the study of MAXCUT by considering cut polytopes of graphs not containing a specific minor.

Organization of this chapter. In Chapter 2.1, we consider $K_{3,3}$ -minor-free graphs. Complementing the results on K_5 -minor-free graphs, we provide the full linear description of cut polytopes of $K_{3,3}$ -minor-free graphs.

Moreover, we give an algorithm solving MAXCUT on $K_{3,3}$ -minor-free graphs, requiring only the running time for MAXCUT on planar graphs. This is somewhat surprising, as K_5 -minor-free graphs admit an easier linear description, while we achieve a better running time for MAXCUT on $K_{3,3}$ -minor-free graphs.

Starting the investigation of geometric properties of cut polytopes, in Chapter 2.2 we completely characterize graphs that provide a simple or simplicial cut polytope. In particular, it turns out that graphs providing a simple cut polytope are precisely the C_4 -minor-free graphs. The simplicial case can only occur for finitely many graphs.

An aim of this chapter is more in putting together some loose ends in the literature than providing completely novel results. Several results are in fact rather expected and some proofs are rather straight forward. However, we emphasize that

there are some pitfalls one can easily run into, e.g.—as explained in Chapter 2.1—Theorem 2.1.7 cannot be deduced just from a graph decomposition into 2-sums, as has been previously suggested. As such, this chapter is also to be understood as a collection and service.

2.1 $K_{3,3}$ -minor-free Graphs

In this section, we consider $K_{3,3}$ -minor-free graphs and provide the complete linear description of their cut polytopes. We also show that this yields an efficient algorithm for MAXCUT on $K_{3,3}$ -minor-free graphs. This complements the known facts on K_5 -minor-free graphs. Moreover, since K_5 is maximal $K_{3,3}$ -minor-free but not weakly bipartite, we obtain the first full polyhedral description of a general minor-closed graph class apart from weakly bipartite graphs.

We first characterize maximal $K_{3,3}$ -minor-free graphs. Per se, this is not new: it is sometimes referenced to (different papers by) Wagner (see for example [Die90] without a proof); a complete proof in modern terminology was given in [Tho99]. Here, we propose a slightly different approach, using 3-connectivity components. This provides a simpler, more basic proof and turns out to be directly usable for our polyhedral and our algorithmic results.

Let $G = (V, E)$ be a 2-connected, not necessarily simple graph and let $\{v, w\}$ be a *split pair* in G , i.e., $G - \{v, w\}$ is disconnected or there are parallel edges connecting v and w . The *split classes* of $\{v, w\}$ are given by a partition E_1, \dots, E_k of E such that two edges are in a common split class if and only if there is a path between them neither containing v nor w as an internal node. As G is 2-connected, it is easy to see that v and w are both incident to each split class. For a split class C let $\bar{C} = E \setminus C$. A *Tutte split* replaces G by the two graphs $G_1 = (V(C), C \cup \{e\})$ and $G_2 = (V(\bar{C}), \bar{C} \cup \{e\})$, provided that $G_1 - e$ or $G_2 - e$ remains 2-connected. Thereby, e is a new *virtual* edge connecting v and w ; the other edges are called *original*. Observe that this operation may yield parallel edges. Iteratively splitting the graphs via Tutte splits gives the unique *3-connectivity decomposition* of G . Its components, the so-called *skeletons* can be partitioned into the following sets: a set S of cycles, a set P of edge bundles (two nodes joined by at least 3 edges), and a set R of 3-connected graphs. See, e.g., [Tut66, HT73].

Lemma 2.1.1. *Any maximal $K_{3,3}$ -minor-free graph G is 2-connected.*

Proof. Clearly, G is connected, as otherwise we could join two connected components via an edge without obtaining a $K_{3,3}$ -minor. Assume that G is not 2-connected and let $v \in G$ be a cut-node separating G into G_1 and G_2 . Choose $w_1 \in G_1$ and $w_2 \in G_2$ adjacent to v and obtain the graph \tilde{G} from G by adding the edge w_1w_2 . As a side note, this operation retains planarity for planar G . Since

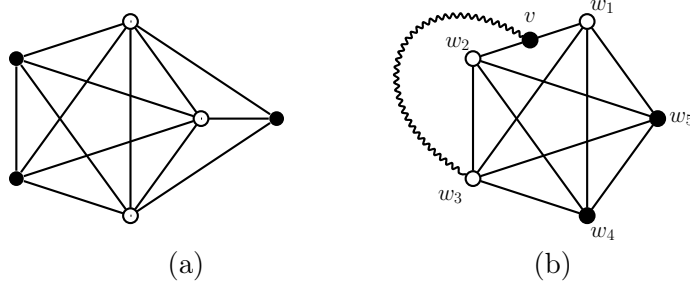


Figure 2.1: The graphs of the proof of Proposition 2.1.2

\tilde{G} contains only two paths between G_1 and G_2 but $K_{3,3}$ is 3-connected, \tilde{G} is still $K_{3,3}$ -minor-free. This contradiction concludes the proof. \square

Proposition 2.1.2. *Let G be a maximal $K_{3,3}$ -minor-free graph. Then, G can be decomposed as a strict clique-sum $G = G_1 \oplus_2 \cdots \oplus_2 G_k$, where each G_i is either a planar triangulation or a copy of K_5 .*

Proof. Let G be a maximal $K_{3,3}$ -minor-free graph. By Lemma 2.1.1, G is 2-connected, so we may consider its 3-connectivity decomposition. Whenever a virtual edge ab was introduced, both parts of the Tutte split contain a path between a and b . Furthermore, G contains a $K_{3,3}$ -minor if and only if one of the components of its decomposition does. But then, if G would not contain an original edge connecting a and b , we could introduce it without creating a $K_{3,3}$ -minor. Thus, each virtual edge corresponds to an edge $e \in E(G)$ by maximality of G , and G is the strict 2-sum of cycles and 3-connected graphs. By maximality of G , the cycles are triangles, a trivial form of a planar triangulations.

Let H be a 3-connected graph from this sum. If H is planar, then – by maximality – it is a triangulation. Otherwise, by Kuratowski’s Theorem, H contains a K_5 -subdivision. Assume that $H \neq K_5$. If H contains K_5 as a subgraph, then it contains the graph shown in Figure 2.1((a)) as a minor, and thus a $K_{3,3}$ -minor, which yields a contradiction.

Assume that H contains a proper K_5 -subdivision with Kuratowski nodes $S = \{w_1, \dots, w_5\}$ and let $v \in V(H) \setminus S$ be a node of this subdivision. Since H is 3-connected, there are disjoint paths from v to three pairwise distinct Kuratowski nodes, say w_1, w_2, w_3 . But then H contains the graph of Figure 2.1((b)) as a minor, which itself contains a $K_{3,3}$ -minor. This concludes the proof. \square

Proposition 2.1.2 allows us to classify all facets of cut polytopes of maximal $K_{3,3}$ -minor-free graphs:

Theorem 2.1.3. *Let G be a maximal $K_{3,3}$ -minor-free graph. Then, all facets of $\text{CUT}^\square(G)$ are given by cycle inequalities for each induced cycle in G and switchings*

of the facet-defining inequality

$$\sum_{e \in E(K_5)} x_e \leq 6 \quad \text{for each } K_5\text{-subgraph.} \quad (2.1)$$

Proof. We know from Theorem 1.5.1 that the facets of the cut polytope of a 2-sum of graphs are given by taking all facets of the cut polytopes of both graphs and identifying common variables. Moreover, by Theorem 1.5.6 all facets of the cut polytope of a planar triangulation are given by cycle inequalities; the facets of $\text{CUT}^\square(K_5)$ are given by metric inequalities and switchings of (2.1) [DL10, Chapter 30.6]. Since maximal $K_{3,3}$ -minor-free graphs are 2-sums of copies of K_5 and planar triangulations, this yields the claimed result. \square

We can use Theorem 2.1.3 to classify the facets of the cut polytope of *any* $K_{3,3}$ -minor-free graph.

Corollary 2.1.4. *Let G be a $K_{3,3}$ -minor-free graph. Then, G can be decomposed as a (not necessarily strict) k -sum of planar graphs and/or copies of K_5 , with $k = 1, 2$.*

Let H be a maximal $K_{3,3}$ -minor-free graph containing G . Then, the facets of $\text{CUT}^\square(G)$ are obtained by projecting $\text{CUT}^\square(H)$ onto $\{x_e = 0 : e \in E(H) \setminus E(G)\}$.

Proof. The decomposition claim follows from Proposition 2.1.2. Alternatively, we can obtain G from H by deleting edges. On the level of cut polytopes, the effect of an edge deletion $e \in E(H)$ corresponds to a projection onto $\{x \in \mathbb{R}^E : x_e = 0\}$. \square

On the level of facets, a projection of a polytope to a coordinate hyperplane is given by eliminating variables. This can be done by Fourier-Motzkin elimination (see Theorem 1.2.2), which is made more precise in the following example.

Example 2.1.5. Consider the non-maximal $K_{3,3}$ -minor-free graph G shown in Figure 2.2. It is obtained by taking the non-strict 2-sum of two copies of K_5 . Let these copies of K_5 be $G_1 = (V, E)$ and $G_2 = (W, F)$ with vertex sets $V = \{v_1, v_2, v_3, u_1, u_2\}$ and $W = \{u_1, u_2, w_1, w_2, w_3\}$.

Both $G_1 - u_1u_2$ and $G_2 - u_1u_2$ are planar and chordal. Thus, all facets of their cut polytopes are given by metric inequalities, and those are also facets of $\text{CUT}^\square(G)$. All other facets of $\text{CUT}^\square(G)$ are obtained by taking a pair of facets \mathcal{F}_1 of $\text{CUT}^\square(G_1)$ and \mathcal{F}_2 of $\text{CUT}^\square(G_2)$ and eliminating the variable $x_{u_1u_2}$ by summing the corresponding inequalities. In the following we focus on the latter class of facets. Choosing one representative for each class of facet-defining inequalities of G_1 and G_2 we get:

$$(1) \text{ one metric inequality of } G_1: \quad x_{u_1u_2} + x_{u_2v_1} + x_{u_1v_1} \leq 2,$$

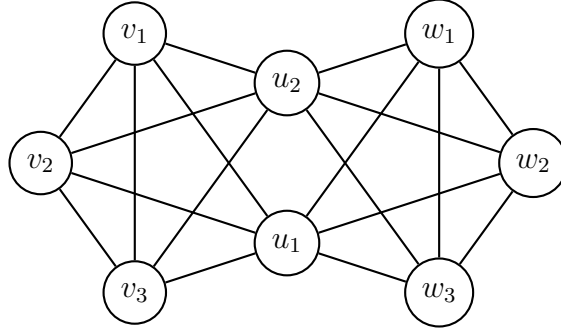


Figure 2.2: The graph of Example 2.1.5

- (2) one hypermetric inequality of G_1 : $\sum_{e \in E} x_e \leq 6$,
- (3) one metric inequality of G_2 : $-x_{u_1 u_2} - x_{u_2 w_1} + x_{u_1 w_1} \leq 0$,
- (4) one hypermetric inequality of G_2 : $\sum_{f \in F: u_2 \notin f} x_f - \sum_{f \in F: u_2 \in f} x_f \leq 2$.

Using Fourier-Motzkin elimination we have to sum each pair of inequalities such that there is one facet of each graph:

$$\begin{aligned} x_{u_2 v_1} + x_{u_1 v_1} - x_{u_2 w_1} + x_{u_1 w_1} &\leq 2, & (1+3) \\ \sum_{f \in F: u_2 \notin f} x_f - x_{u_2 w_1} - x_{u_2 w_2} - x_{u_2 w_3} + x_{u_2 v_1} + x_{u_1 v_1} &\leq 4, & (1+4) \\ \sum_{e \in E: e \neq u_1 u_2} x_e - x_{u_2 w_1} + x_{u_1 w_1} &\leq 6, & (2+3) \\ \sum_{e \in E: e \neq u_1 u_2} x_e + \sum_{f \in F: u_2 \notin f} x_f - x_{u_2 w_1} - x_{u_2 w_2} - x_{u_2 w_3} &\leq 8. & (2+4) \end{aligned}$$

(1+3) is a cycle inequality. Switching (1+4) at $\delta(\{u_2\})$ shows that this inequality is equivalent to (2+3). These inequalities correspond to copies of K_5 with one subdivided edge contained in G . The support graph of facet (2+4) is G : This type of inequality is neither facet-defining for complete graphs nor does it belong to one of the mentioned classes of facet-defining inequalities in Chapter 1.4. \blacktriangleleft

As demonstrated in the above example, Fourier-Motzkin elimination yields all facet-defining inequalities of a non-maximal $K_{3,3}$ -minor-free graph as sums of cycle inequalities and hypermetric K_5 -inequalities. From Theorem 1.5.6 we can thus deduce that the support graph of a facet-defining inequality is an edge, a cycle or contains a K_5 -minor. Considering the sum of two facets \mathcal{F}_1 and \mathcal{F}_2 used to eliminate the variable x_e we observe the following: If \mathcal{F}_2 is a cycle inequality,

summing it to \mathcal{F}_1 acts on the support graph of \mathcal{F}_1 as subdividing e ; the effect of subdividing an edge in the support graph of a facet is described in Theorem 1.5.9. If \mathcal{F}_2 is a hypermetric K_5 -inequality, summing it to \mathcal{F}_1 acts on the support graph of \mathcal{F}_1 as replacing e by $K_5 - e$; all non-zero coefficients of the obtained inequality are ± 1 . Although possible, it is tedious to determine the exact signs and thus the constant term of the inequalities. However, we can concisely describe the facets' support graphs. For this, recall that each cycle is a subdivision of a triangle.

Corollary 2.1.6. *Let \mathcal{F} be a facet of the cut polytope of a $K_{3,3}$ -minor-free graph G . All its non-zero coefficients are ± 1 and its support graph is an induced subgraph of G that is either*

- *an edge that is not contained in a triangle, or*
- *obtained from a triangle or a K_5 by repeatedly (possibly zero times) subdividing edges and/or replacing an edge e by $K_5 - e$.*

Algorithmic Consequences. Barahona [Bar83, Section 4] gave an $\mathcal{O}(|V|^4)$ algorithm for MAXCUT on K_5 -minor-free graphs. We complement this result by giving an algorithm for MAXCUT on $K_{3,3}$ -minor-free graphs whose running time is identical to that of planar MAXCUT is given. Currently, the best known running time for the latter is $\mathcal{O}(|V|^{\frac{3}{2}} \log |V|)$ [LP12, SWK90].

Our algorithm is based on the decomposition from Proposition 2.1.2. Considering only the case of two subgraphs, joined via a 2-sum, either their common vertices are in the same partition side or not. A straight forward idea to compute a maximum cut in this case would be to compute both cases for both subgraphs and pick the best choice. However, the number of subgraphs occurring in a graphs decomposition into 2-sums may be in $\Theta(|V(G)|)$. Thus, the described procedure ad hoc yields exponential running time. Besides solving this issue, we discuss an efficient, in particular even linear-time, procedure to find and utilize such a decomposition of a given graph.

We use a data structure to efficiently consider the components of the 3-connectivity decomposition of G . Recall that they are cycles S , edge bundles P , and 3-connected graphs R . The *SPR-tree* $T = T(G)$ has a node for each element of S , P , and R [dBT96, CH17]¹. For a node $v \in V(T)$, let H_v denote its corresponding skeleton. Two nodes $v, w \in V(T)$ are adjacent if and only if H_v and H_w share a virtual edge. G can be reconstructed from T by taking the non-strict 2-sum of components whenever their corresponding nodes are adjacent in T . Following this interpretation, P -nodes containing a non-virtual edge represent strict 2-sums of

¹The data structure is also known as *SPQR-tree*. However, the originally proposed nodes of type Q (as well as the tree's orientation) have often turned out to be superfluous.

their adjacent components of the decomposition. T has only linear size and can be computed in $\mathcal{O}(|E(G)|)$ time [HT73, Lemma 15].

Theorem 2.1.7. *The MAXCUT problem on $K_{3,3}$ -minor-free graphs can be solved in the same time complexity as MAXCUT on planar graphs.*

Proof. Let $G = (V, E)$ be a $K_{3,3}$ -minor-free graph with edge weights c_e , $e \in E$. Let $p(n) \in \Omega(n)$ be the best known running time for MAXCUT on planar graphs with n nodes. For $A, B \subseteq E$ we denote by $\beta_G(A, B)$ the maximum weight over cuts $\delta \subseteq E(G)$ with $A \subseteq \delta$ and $B \cap \delta = \emptyset$. If G is not 2-connected, we apply the algorithm to its 2-connected components (which can be identified in linear time). Assume in the following that G is 2-connected.

We want to insert “original” edges of weight 0 into G between split pairs corresponding to Tutte splits. This will allow us to only consider strict 2-sums. To this end compute the SPR-tree $T = T(G)$. For any P -node $v \in V(T)$ whose H_v contains only virtual edges, introduce a new original edge of weight 0 into H_v , and therefore also into G . For any adjacent non- P -nodes $v, w \in V(T)$, let ab be the virtual edge shared between their components. We introduce a new original edge ab into G . This yields a new P -node u subdividing the edge vw in T . The edge bundle H_u contains the new original edge together with two virtual edges, one shared with H_v , the other with H_w . By this construction, for every virtual edge there is an original edge with the same end nodes. Throughout the following, we always consider the weight of a virtual edge ab to be identical to the weight of the original edge ab . We continue to denote the resulting graph and tree by G and T , respectively.

Let v be a leaf in T and ab be the virtual edge contained in $H = H_v$. Note that v is either an S - or an R -node and thus, H is either a copy of K_5 or planar. We compute $\beta^+ = \beta_H(\{ab\}, \emptyset)$ and $\beta^- = \beta_H(\emptyset, \{ab\})$. If $H = K_5$, this requires only constant time. Thus the needed work is bounded by $\mathcal{O}(p(|V(H)|))$. Let $\gamma = \beta^+ - \beta^-$ be the gain/loss by having ab in the cut, respectively. Removing v from T and therefore all edges of H_v from G yields a graph G' . $T(G')$ is obtained from $T - v$ by removing the potential P -node-leaf (and considering the “dangling” virtual edge as original, retaining its current cost). Setting the cost of the *original* edge ab to γ (after the computation of β^+ and β^-) yields that the maximum cut on G is exactly $\beta^- + \xi$, where ξ is the maximum cut in G' (after updating the edge weight).

In this way, we can iteratively compute a maximum cut on G by eliminating all nodes of its SPR-tree. The SPR-tree of G can be built in $\mathcal{O}(|E|)$ time. Let H_1, \dots, H_k be the components of G corresponding to R - and S -nodes in $T(G)$, $k \leq |V|$. By planarity (or constant size of K_5), we have $|E(H_i)| \in \mathcal{O}(|V(H_i)|)$, and hence $|E| \in \mathcal{O}(|V|)$. For each $i \in [k]$, we require only $\mathcal{O}(p(|V(H_i)|))$ time for the

computations on H_i . Since $p(|V|) \in \Omega(|V|)$ we have $\sum_{i=1}^k p(|V(H_i)|) \in \mathcal{O}(p(|V|))$. The claim follows. \square

2.2 Simple and Simplicial Cut Polytopes

In this section, we completely characterize graphs whose cut polytopes are simple or simplicial.

In [Gan13], it was claimed that $\text{CUT}^\square(G)$ is simple if and only if G contains no C_4 -minor. Unfortunately, the given proof has some gaps. For example, [Gan13, Proposition 3.2.4.] claims that a 0-1-polytope is simple if and only if it is smooth. The proof mistakenly assumes that $\text{CUT}^\square(G)$ is always the polytope corresponding to the cut-variety in the sense of toric geometry. It is then used that a toric variety is smooth if and only if the corresponding polytope is, see [CLS11, Theorem 2.4.3]. However, the cut polytope $\text{CUT}^\square(K_3)$ is simple but not smooth, since the edges $(1, 1, 0)$, $(1, 0, 1)$ and $(0, 1, 1)$ do not form a basis of \mathbb{Z}^3 . Contrarily the cut variety of K_3 is smooth, see [SS08, Corollary 2.4].

Nevertheless, in the following we show that the claimed characterization of graphs whose cut polytopes are simple is true. Our proof only requires basic tools from graph theory and discrete geometry.

Definition 2.2.1. An *ear* in a graph G is a maximal path whose internal nodes have degree 2 in G . An *ear decomposition* of a 2-connected graph G is a decomposition $G = \bigcup_{i=0}^n G_i$ such that G_0 is a cycle and G_k is an ear of $\bigcup_{i=0}^k G_i$ for all $1 \leq k \leq n$.

A graph is 2-connected if and only if it admits an ear decomposition, see, e.g., [Die18, Proposition 3.1.2]. Utilizing this, we can characterize C_4 -minor-free graphs in terms of decompositions. This is not new, as it is stated (without a proof) in [Die90]. We include a short proof for the convenience of the reader.

Lemma 2.2.2. *Let G be a connected graph. Then, the following are equivalent:*

- (i) G is C_4 -minor-free;
- (ii) $G = G_1 \oplus_1 \cdots \oplus_1 G_k$ with $G_i = K_2$ or $G_i = K_3$ for each $i \in [k]$.

Proof. Since K_2 and K_3 are C_4 -minor-free and 1-sums create cut-nodes, it is easy to see that (ii) implies (i). To show the reverse direction, let G be a C_4 -minor-free graph. Considering its 2-connected components gives a decomposition $G = G_1 \oplus_1 \cdots \oplus_1 G_k$, where $G_i = K_2$ or G_i is 2-connected.

It is left to show that the only 2-connected C_4 -minor-free graph is K_3 . Assume that G is a 2-connected C_4 -minor-free graph and consider its ear-decomposition $G = G_0 \cup \cdots \cup G_k$. Since G is C_4 -minor-free, G_0 is a copy of K_3 . Attaching an ear to two of its nodes would yield a C_4 -minor. Hence $G = K_3$. \square

Given this characterization, we are able to show that C_4 -minor-free graphs are exactly those graphs whose cut polytopes are simple.

Theorem 2.2.3. *The following are equivalent:*

- (i) $\text{CUT}^\square(G)$ is simple;
- (ii) $\text{CUT}^\square(G)$ is the product of $(0,1)$ -simplices arising as the cut polytopes of the 2-connected components of G , which then necessarily have to be K_2 or K_3 .
- (iii) G is C_4 -minor-free.

Observe that the equivalence of (i) and (ii) can be seen as the cut version of the structure of simple $(0,1)$ -polytopes according to [KW00]: A $(0,1)$ -polytope is simple if and only if it is the product of $(0,1)$ -simplices. However, it is not a priori clear that the latter simplices are cut polytopes; even if they are, it is unclear how the corresponding graphs are related to G . We hence need to explicitly prove Theorem 2.2.3.

Proof. Note that a product of polytopes is simple if and only if each of the polytopes is simple. If G is not connected, then $\text{CUT}^\square(G)$ is the product of the cut polytopes of the connected components of G . If G is connected but not 2-connected, it can be decomposed as $G = G_1 \oplus_1 \dots \oplus_1 G_k$ such that G_i is either 2-connected or a copy of K_2 . Recall that $\text{CUT}^\square(K_2)$ and $\text{CUT}^\square(K_3)$ are simplices.

For the equivalence of (i) and (ii), it remains to show that the cut polytope of a 2-connected graph G is not simple if $G \neq K_3$. Then, each edge $e \in E(G)$ is contained in a chordless cycle C_e . By Theorem 1.5.6 the inequalities

$$x_e - \sum_{f \in E(C_e) \setminus \{e\}} x_f \leq 0 \quad (2.2)$$

define $|E(G)|$ many different facets of $\text{CUT}^\square(G)$ that contain the origin.

If $G = C_n$, $n \geq 4$, no edge is contained in a triangle and $x_e \geq 0$ defines a facet of $\text{CUT}^\square(G)$ for all $e \in E$. Hence, 0 is contained in at least $2|E(G)|$ many different facets and as $\dim(\text{CUT}^\square(G)) = |E(G)|$, the cut polytope is not simple. Similarly, if $G \neq C_n$, then there has to exist a chord e in some cycle. In particular, e lies in two chordless cycles. Thus, the origin is contained in at least $|E| + 1$ facets and hence $\text{CUT}^\square(G)$ is not simple. Part (iii) is equivalent to (ii) by Lemma 2.2.2. \square

Next we study graphs whose cut polytopes are simplicial. It was shown in [DL10] that the cut polytope of K_n is not simplicial for $n \geq 5$. We generalize this result by giving a complete characterization of graphs with simplicial cut polytopes.

Table 2.1: All graphs on $n \leq 4$ non-isolated nodes (cf. Theorem 2.2.4)

Graph G										
$\text{CUT}^\square(G)$ simplicial?	✓	✓	✓	✗	✗	✓	✓	✗	✗	✓

Theorem 2.2.4. *Let G be a graph with no isolated nodes. Then, the following are equivalent:*

- (i) $\text{CUT}^\square(G)$ is simplicial;
- (ii) G is one of the following graphs:

$$K_2, K_2 \cup K_2, K_2 \oplus_1 K_2, K_3, K_4, C_4.$$

Proof. Recall that if H is obtained from G by edge contraction, $\text{CUT}^\square(H)$ is a face of $\text{CUT}^\square(G)$. Hence, if $\text{CUT}^\square(H)$ is not a simplex, $\text{CUT}^\square(G)$ is not simplicial.

First consider the graphs in Table 2.1. Recall that $\text{CUT}^\square(K_2)$ is simplicial and 1-dimensional. By (1.1) and (1.2), this yields that $\text{CUT}^\square(K_2 \cup K_2)$ and $\text{CUT}^\square(K_2 \oplus_1 K_2)$ are simplicial. $\text{CUT}^\square(K_3)$ is a 3-simplex. It is straight-forward to verify that $\text{CUT}^\square(K_4)$ is affine isomorphic to the 6-dimensional cyclic polytope on 8 vertices and $\text{CUT}^\square(C_4)$ is affinely isomorphic to a cross-polytope, both of which are simplicial. The remaining graphs are all contractible to a path of length 3 and thus, their cut polytopes are not simplicial.

If G is a connected graph on at least 5 vertices, G is contractible to a connected graph on 4 vertices. None of those yield a simplex. If G is disconnected but not $K_2 \cup K_2$, then G is contractible to the disjoint union of copies of K_2 . Since the cut polytope of the latter is a hypercube, $\text{CUT}^\square(G)$ is not simplicial. \square

2.3 Open Problems

We have determined the linear description of cut polytopes of $K_{3,3}$ -minor-free graphs and classified all graphs with a simple or simplicial cut polytope.

Throughout this chapter one can see that besides graph minors, the decomposition of graphs into clique-sums of specific graphs is a useful tool to understand cut polytopes. This motivates several questions discussed in the following.

Open Problem 2.3.1. Can one give the linear description of cut polytopes of H -minor-free graphs, for single-crossing graphs $H \neq K_5, K_{3,3}$?

By Theorem 1.5.1 for $k \leq 3$ the linear description of a strict k -sum of two graphs is given by taking all facet-defining inequalities of both graphs and identifying common variables. This can be traced back to the fact that in these cases

$\text{CUT}^\square(K_k)$ is a simplex. Although this does not hold for $k \geq 4$, the cut polytope of K_4 is a cyclic polytope and as such well understood. Therefore, the following question arises:

Open Problem 2.3.2. Can one give a linear description of the 4-sum of two graphs in terms of their linear descriptions?

While we give a linear description of cut polytopes of $K_{3,3}$ -minor-free graphs in Chapter 2.1, there are further graphs that fall under the same facet regime (interestingly, even $K_{3,3}$ itself). We thus ask:

Open Problem 2.3.3. Can one characterize all graphs whose cut polytopes are described by the inequalities from Chapter 2.1?

Chapter 3

Maximum Cut Parameterized by Crossing Number

This chapter is based on [CDJ⁺20].

A graph is *1-planar* if it allows a drawing where each edge is involved in at most one crossing. A *1-plane* graph is such a graph, together with an embedding realizing this property. The MAXCUT problem on 1-plane graphs with k edge crossings has recently been shown to be *fixed-parameter tractable* (FPT) with parameter k [DKM18]. More precisely, it was shown that such instances can be solved in $\mathcal{O}(3^k \cdot p(n))$ time, where $p(n)$ is the running time of a polynomial-time MAXCUT algorithm on planar graph with n nodes, e.g., [LP12, SWK90]. There are no restrictions on the edge weights. In this chapter we improve these in several ways: Firstly, we drop the requirement of 1-planarity, i.e., we consider graphs that can be drawn with at most k crossings (even if multiple such crossings lie on the same edge). We therefore handle the case of the well-established notion of the graph's *crossing number*. Secondly, we reduce the runtime dependency on k from 3^k to 2^k . Finally, unlike the previous result, our approach can be extended to an FPT algorithm which does not even require a crossing-realizing drawing as an input; however, this increases the running time and requires a deep algorithm from the literature as a black box [Gro04, KR07]. Interestingly, we achieve these results by a *simpler* approach (compared to [DKM18]). Comparing our algorithm with [GLV01], we have no restrictions on the edge weights. Even in the restricted scenario, our algorithm is faster for graphs whose crossing number is at most twice its genus. Furthermore, we require only easily-implementable data structures and subalgorithms (if we are given a crossing-realizing drawing), compared to advanced methods from algebra.

Organization of this chapter. In Chapter 3.1 we provide some background on the crossing number of a graph. In Chapter 3.2, we present our new algorithm and prove its correctness and running time. The general idea of this algorithm is to recursively get rid of each crossing, each time resulting in two new subinstances. We end up with a set of up to 2^k planar graphs, each of which can be solved using a known polynomial-time MAXCUT algorithm for planar graphs. The maximum over all these subinstances then yields a maximum cut in the original instance.

Finally in Chapter 3.3, we consider parameterizing the problem by the *minor* crossing number (see below for details). This measure is always at most the graph's crossing number. While the exponential dependency on the respective parameter is identical, the running time only slightly increases in its polynomial part.

Independent work. Kobayashi et al. [KKMT19b] independently and simultaneously obtained another fixed-parameter tractable algorithm for MAXCUT parameterized by crossing number achieving the same running time. However, while we can always stay in the realm of maximum cuts when solving subinstances, they have to consider maximum weighted b -factor problems. Their preprint was uploaded to arXiv shortly after ours [CDJ⁺19, KKMT19a].

3.1 Preliminaries

Recall that a graph is *planar* if it admits a drawing without any crossings. It is well-known that planarity can be tested in linear time [HT73]. For non-planar graphs it is natural to ask for a drawing with as few crossings as possible. The smallest such number is the *crossing number* $\text{cr}(G)$ of G . Not only is it NP-hard to compute $\text{cr}(G)$ [GJ83], but even the so called *realizability* problem turns out to be NP-hard [Kra91]: Given a graph G and a set X of edge pairs, is there a drawing \mathcal{D} of G such that X contains an edge pair if and only if the pair's two edge curves cross in \mathcal{D} ? The key problem in testing realizability is that it is hard to figure out whether there exist *orderings* of the crossings along the respective edges that allow the above properties.

Therefore, sometimes more restricted crossing variants are considered. For example, *1-planar* graphs admit drawings where every edge is involved in at most one crossing. Not all graphs can be drawn in such a way, since 1-planar graphs can have at most $4|V| - 8$ edges; also, the 1-planar number of crossings is in general larger than $\text{cr}(G)$ [PT97, CKMV19].

For a general drawing (not necessarily 1-plane), we typically encode its crossings as a *crossing configuration* \mathcal{X} . Therein, we not only store the pairs of edges that cross, but for each edge also the order of the crossings as they occur along its curve. The feasibility of a crossing configuration can be tested in time linear in

$|V| + |\mathcal{X}|$ by replacing crossings with dummy nodes of degree 4, testing planarity, and checking the cyclic order around dummy nodes.¹ Although we will not require this fact in the following, this also allows us to efficiently deduce a drawing that respects \mathcal{X} . It is well understood that we can restrict ourselves to *good drawings* when considering the (traditional) crossing number of graphs: adjacent edges never cross and no edge pair crosses more than once.

3.2 Algorithm

Our main idea for computing the maximum cut in an embedded weighted graph is to eliminate its crossings one by one. In the end, we use a MAXCUT algorithm for planar graphs. We first introduce a slight variant of MAXCUT:

Definition 3.2.1 (Partially-Fixed Maximum Cut, PF-MAXCUT). Given an edge weighted graph $G = (V, E, c)$ and a set of *fixed* edges $F \subseteq E$, find a cut of maximum value that contains all elements of F .

A cut is *feasible* if it contains F . A PF-MAXCUT instance is *infeasible* if it does not allow a feasible cut. It is easy to see that an instance is infeasible if and only if F contains a cycle of odd length. We denote a maximum objective value by $\text{MaxCut}_{\text{pf}}(G, F)$, and let $\text{MaxCut}_{\text{pf}}(G, F) = -\infty$ for infeasible instances.

Observe that (as for MAXCUT) we do not need to consider a given crossing configuration \mathcal{X} as part of the problem description (see Corollary 3.2.7). However, since having \mathcal{X} allows for simplifications and a better running time, we will for now assume that we are given the graph together with a crossing configuration \mathcal{X} .² We will explain later how to remove this assumption in Corollary 3.2.7.

Given any edge vw with weight c_{vw} in a PF-MAXCUT instance, we define the operation to *bisubdivide* vw at v as follows: Subdivide vw twice, i.e., replace vw by a path of length 3 with two new degree-2 nodes. We denote the new node incident to v or w by \bar{v} or \bar{w} , respectively. We consider the notation $\bar{\cdot}$ an operand.³

The edges $v\bar{v}$ and $\bar{v}\bar{w}$ have weight 0, $\bar{w}w$ retains the weight c_{vw} . Furthermore, we add $v\bar{v}, \bar{v}\bar{w}$ to F , and if $vw \in F$, we replace it in F by $\bar{w}w$. Clearly, both $v\bar{v}, \bar{v}\bar{w}$

¹In general, a specified edge pair may not really cross but merely “touch”; this is trivial to detect after testing planarity by checking the cyclic order of the edges around the dummy node. Given such a “flawed” configuration, we trivially obtain one with less crossings by removing such crossing pairs from \mathcal{X}

²As noted above, we may assume that no such \mathcal{X} ever specifies “touching points”; we can reduce such configurations whenever our algorithm retrieves a new crossing configuration.

³Observe that per recursion step, we will bisubdivide at most one edge per incident node (recall that adjacent edges never cross in good drawings). Thus, the above simple notation is unambiguous. In the graphs of the subproblems, see below, we may assume the nodes to be named afresh, and thus we may again perform bisubdivisions without creating notational ambiguity.

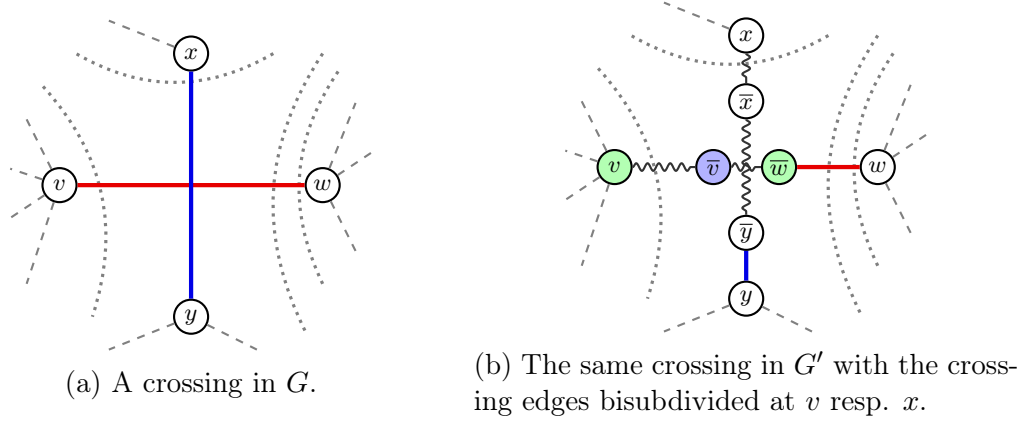


Figure 3.1: The situation at a crossing between vw and xy in G . In G' , the two edges of the crossing are bisubdivided at v and x , respectively, and the zig-zag edges are added to the set of fixed edges F' . As an example, the node coloring at v, \bar{v}, w gives a partition of these nodes that is forced by the respective newly added edges in F' . (Dashed and dotted edges show examples of other edges in G , resp. G' .)

will be in any feasible cut; node \bar{w} will always lie in the same partition set as v , and \bar{v} in the other (cf. Figure 3.1(b)). Most importantly this gives:

Lemma 3.2.2. *The feasible cuts in an original PF-MAXCUT instance $\langle G, F \rangle$ are in 1-to-1 correspondence to feasible cuts of equal value in a bisubdivided instance $\langle G', F' \rangle$.*

Proof. Let vw be the edge in G that is bisubdivided at v to obtain $\langle G', F' \rangle$. By construction, we know that both edges $v\bar{v}$, $\bar{v}w$ have cost 0 and are in F' , and thus in any F' -feasible cut. Consequently, in any F' -feasible cut, v and \bar{w} will lie in a common partition set. Let $S' \subset V(G')$ be a node subset that induces some feasible (with respect to F') cut in G' . Then, the node set $S = S' \setminus \{\bar{v}, \bar{w}\}$ induces a feasible (with respect to F) cut in G . Cut $\delta(S)$ contains edge vw if and only if $\delta(S')$ contains $\bar{w}w$. Since both these edges have identical cost, the total costs of both cuts are equal.

Inversely, let $S \subset V(G)$ be a node subset that induces some feasible (with respect to F) cut in G . Then, consider the cut in G' induced by $S' = S \cup \{s\}$, where $s = \bar{w}$ if $v \in S$, and $s = \bar{v}$ otherwise. Both fixed edges $v\bar{v}$, $\bar{v}w$ are in $\delta(S')$ and the cut is thus feasible. Again, $\delta(S')$ contains edge $\bar{w}w$ if and only if $\delta(S)$ contains vw , and both cut values are thus equal. \square

When we *identify* two nodes a, b in a graph with one another, they become a common entity that is incident to all of their former neighbors. We will only

identify nodes that are neither adjacent nor share neighbors.

When identifying nodes in G of some PF-MAXCUT instance $\langle G, F \rangle$, the set F is retained, subject to replacing the edges formerly incident to a or b with their new counterparts.

We are now ready to describe our algorithm. We are given a MAXCUT instance $G = (V, E, c)$, together with some crossing configuration \mathcal{X} with k crossings. Let $F = \emptyset$ be the set of fixed edges and consider $\langle G, F \rangle$ as a PF-MAXCUT instance. From $\langle G, F, \mathcal{X} \rangle$, we pick a crossed edge vw , and derive two new triplets $T_i = \langle G_i, F_i, \mathcal{X}_i \rangle$, for $i \in \{v, w\}$. Both derived crossing configurations \mathcal{X}_i attain at most $k - 1$ crossings and we can call our algorithm recursively on T_v and T_w . As a base case, the derived graphs become planar and (after a preprocessing to deal with the fixed edges) we apply an efficient MAXCUT algorithm for planar graphs. The solutions of $\langle G_i, F_i \rangle$, for $i \in \{v, w\}$, yield a solution of $\langle G, F \rangle$. Observe, however, that $\langle G_i, F_i \rangle$ may become infeasible.

Let us describe this recursion step formally (cf. also Figures 3.1 and 3.2). We define the *crossing split* operation that, given a triplet $\langle G, F, \mathcal{X} \rangle$, yields the two triplets T_v and T_w : Let $\langle G = (V, E, c), F \rangle$ be a PF-MAXCUT instance and \mathcal{X} a crossing configuration of G . Consider a crossing $\chi \in \mathcal{X}$ with crossing edges vw and xy . For $j \in \{v, w, x, y\}$, let Y_j be the *ordered sets of crossings* in \mathcal{X} between j and χ (cf. the dotted edges in Figure 3.1: e.g., the crossings between the two dotted edges and $\bar{w}w$ in Figure 3.1(b) are in Y_w as they are between χ and w in Figure 3.1(a)). Let the intermediate instance $\langle G', F' \rangle$ be obtained from $\langle G, F \rangle$ by bisubdividing vw at v and bisubdividing xy at x . For $i \in \{v, w\}$, let $\langle G_i, F_i \rangle$ be the PF-MAXCUT instance obtained from $\langle G', F' \rangle$ by identifying \bar{x} with \bar{i} (see Figures 3.2(c) and 3.2(d)). Intuitively, the two graphs obtained by the identifications represent the two possibilities whether x is on the same side of the cut as v or not. We obtain a corresponding crossing configuration \mathcal{X}_i from \mathcal{X} by removing χ and placing the crossings Y_j (retaining their order) on the edge $j\bar{j}$, for all $j \in \{v, w, x, y\}$. The triplets $T_v = \langle G_v, F_v, \mathcal{X}_v \rangle$ and $T_w = \langle G_w, F_w, \mathcal{X}_w \rangle$ are the results of the crossing split operation with respect to $\langle \chi, vw, xy \rangle$.

Lemma 3.2.3. *Let $\langle G = (V, E, c), F \rangle$ be a PF-MAXCUT instance and \mathcal{X} a crossing configuration of G with k crossings. Let $\chi \in \mathcal{X}$ be any crossing with some crossing edges vw and xy , and consider the crossing split operation with respect to $\langle \chi, vw, xy \rangle$. For $i \in \{v, w\}$, let $\langle G_i, F_i, \mathcal{X}_i \rangle$ be the resulting triplets. Then, we have:*

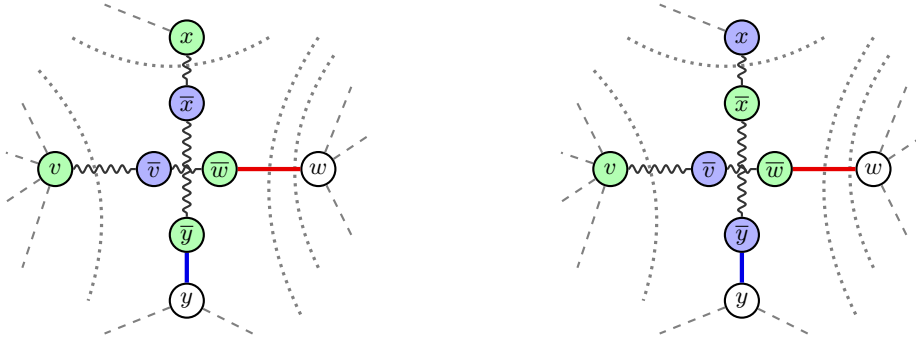
1. *for $i \in \{v, w\}$, \mathcal{X}_i is a feasible crossing configuration for G_i with at most $k - 1$ crossings; and*
2. $\text{MaxCut}_{\text{pf}}(G, F) = \max_{i \in \{v, w\}} \{ \text{MaxCut}_{\text{pf}}(G_i, F_i) \}$.

Proof. Consider any drawing \mathcal{D} of G realizing \mathcal{X} . By routing the new paths $(v\bar{v}, \bar{v}\bar{w}, \bar{w}w$ resp. $x\bar{x}, \bar{x}\bar{y}, \bar{y}y)$ along the curves of their original edges $(vw$ resp. $xy)$ we obtain a drawing \mathcal{D}' of G' from \mathcal{D} . Thereby, for $j \in \{v, w, x, y\}$, we place the new nodes \bar{j} in a close neighborhood of χ on the curve segment between j and χ , so that $\bar{x}\bar{y}$ is only crossed by $\bar{v}\bar{w}$ and vice versa. Note that the number of crossings in \mathcal{D}' is equal to that of \mathcal{D} , since all crossings in Y_j in \mathcal{D} are transferred to the edge $j\bar{j}$ in \mathcal{D}' , for all $j \in \{v, w, x, y\}$, and the original crossing χ between xy and vw in \mathcal{D} has a counterpart χ' in \mathcal{D}' between the edges $\bar{v}\bar{w}$ and $\bar{x}\bar{y}$. Since the edges $\bar{v}\bar{w}$ and $\bar{x}\bar{y}$ are crossing free except for χ' , we can follow (in a close neighborhood) the curves of $\bar{v}\bar{w}$ from any of its end points up to χ' , and onwards from there along the curve of $\bar{v}\bar{w}$ to any of its end points. Since these routes are crossings-free, we call them *free routes*. When we now identify \bar{x} with \bar{v} , we can locally redraw our drawing such that χ' vanishes and no other crossings arise, see Figure 3.2(c). Observe that \bar{x} has precisely two neighbors: \bar{y} and x . The identification is thus such that we may remove \bar{x} and insert edges $\bar{y}\bar{v}$ and $x\bar{v}$ instead. The former can trivially be drawn without any crossings along the free route between \bar{y} and \bar{v} . The curve for the latter edge is the concatenation of the former curve of $x\bar{x}$ and the free route between \bar{x} and \bar{v} . The number of crossings along the edge $x\bar{x}$ (with now $\bar{x} = \bar{v}$) does thus not change. We can perform the analogous redrawing when identifying \bar{x} with \bar{w} , see Figure 3.2(d). This establishes claim (1).

Two nodes v and x can either be on the same side of a cut, or they are on opposite sides. Therefore, we create two new subproblems in which v and x are in the same partition set or not, respectively. In G_v (where we identify \bar{x} with \bar{v}), we have a path of two edges between v and x (namely $v\bar{v}$ and $\bar{v}x$), both of which are in F_v . Thus, v and x have to be in the same partition set, see Figure 3.2(c). Conversely, in G_w (where we identify \bar{x} with \bar{w}), we have a path of three edges between v and x (namely $v\bar{v}$, $\bar{v}\bar{w}$, and $\bar{w}x$), all of which are in F_w . Thus, v and x have to be in different partition sets, see Figure 3.2(d). We can see that the respective constructions do not induce any further restrictions on the set of cuts. In particular, both derived instances still allow any partition choice between w and x , between w and y , and between x and y . Overall, every feasible cut in $\langle G', F' \rangle$ can be realized either in $\langle G_v, F_v \rangle$ or in $\langle G_w, F_w \rangle$.

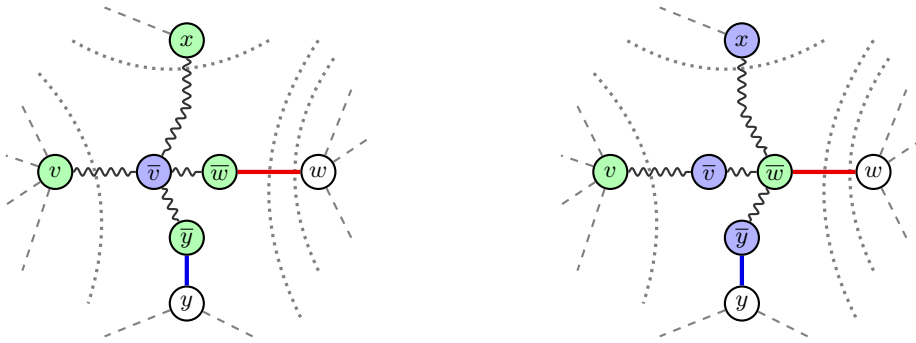
If we know the maximum cut in instance $\langle G_v, F_v \rangle$ and the maximum cut in instance $\langle G_w, F_w \rangle$, we can pick the larger of these two cuts and transfer it back to $\langle G', F' \rangle$. By applying Lemma 3.2.2 twice (once for the bisubdivision of vw at v and once for the bisubdivision of xy at x), the maximum cut in $\langle G', F' \rangle$ induces a maximum cut in $\langle G, F \rangle$ of the same value. Claim (2) follows. \square

If we are in a base case – the considered graph is planar – we can use an efficient MAXCUT algorithm for planar graphs:



(a) Induced partition in G' with x and v on the same side of the partition.

(b) Induced partition in G' with x and v on different sides of the partition.



(c) In G_v , \bar{x} is identified with \bar{v} .

(d) In G_w , \bar{x} is identified with \bar{w} .

Figure 3.2: An illustration of the two cases where v and x are either on the same side of the partition (a/c) or on opposite sides (b/d). In the two graphs G_v and G_w , the crossing was removed while retaining the partition property. The node coloring gives a partition of the nodes that is induced by the newly added edges in F' , resp. F_v or F_w . (Dashed and dotted edges show examples of other edges in G' , resp. G_v or G_w .)

Lemma 3.2.4. *Consider a PF-MAXCUT instance $\langle G = (V, E, c), F \rangle$ with a planar graph G . Let $p(|V|)$ be a polynomial upper bound on the running time of a MAXCUT algorithm on the planar graph G . We can compute an optimal solution to $\langle G, F \rangle$ – or decide that the instance is infeasible – in $\mathcal{O}(p(|V|))$.*

Proof. We transform the PF-MAXCUT instance into a traditional MAXCUT instance by attaching a large weight to the edges in F . Namely, we add M to the weight of each edge $f \in F$, where $M = 2 \cdot \sum_{e \in E} |c_e|$. The omission of a single edge of F from the solution cut (even if picking all other edges of positive weight) will already result in a worse objective value than picking all of F and all edges of negative weight. The instance is infeasible if and only if the computed cut does

not contain all of F ; this can also be deduced purely by checking whether the objective value is at least $M \cdot |F| + \sum_{e \in E: c_e < 0} c_e$. \square

We proved our lemma above for a general case (by adding M to the weight of each edge in F), but in fact we only require a slightly weaker version, since in our algorithm $c_f = 0$ for all $f \in F$. Thus it suffices to set $c_f = M$ instead of adding M to c_f . Using any of the currently fastest MAXCUT algorithms for planar graphs [LP12, SWK90] leads to $\mathcal{O}(|V|^{3/2} \log |V|)$ time in the above lemma. We could speed-up infeasibility detection by checking whether F contains a cycle of odd length prior to the transformation; while this only requires $\mathcal{O}(|V|)$ time via depth-first search, the overall asymptotic runtime for the lemma's claim does of course not improve.

Theorem 3.2.5. *Let $G = (V, E, c)$ be an edge-weighted graph and \mathcal{X} a crossing configuration of G with k crossings. Let $p(n)$ be a polynomial upper bound on the running time of a MAXCUT algorithm on planar graphs with n nodes. We can compute a maximum cut in G in $\mathcal{O}(2^k \cdot p(|V| + k))$ time.*

Proof. As described above, we solve the instance by considering the PF-MAXCUT instance $\langle G, F = \emptyset \rangle$ together with \mathcal{X} . Thus the triplet $\langle G, F, \mathcal{X} \rangle$ forms the initial input of our recursive algorithm \mathcal{R} .

Algorithm \mathcal{R} proceeds as follows on a given triplet: If the triplet's graph is planar, we solve $\langle G, F \rangle$ via Lemma 3.2.4. Otherwise, we use Lemma 3.2.3 to obtain two new input triples T_v, T_w , for each of which we call \mathcal{R} recursively. Their returned solutions (i.p., their solution values) induce the optimum solution for the current input triplet. However, while the number of crossings decreases by (at least) one per recursion step, the graph's size increases by three nodes.

The runtime complexity follows from the fact that we consider two choices per crossing in the given \mathcal{X} , and thus construct 2^k graphs. For each such graph, which has $|V| + 3k$ nodes, we run the planar MAXCUT algorithm. \square

Above, we trivially have $k \in \mathcal{O}(|V|^4)$ and thus $|V| + k \in \mathcal{O}(\text{poly}(|V|))$.

Corollary 3.2.6. *The above algorithm is an FPT algorithm with parameter k , provided that a crossing configuration \mathcal{X} with k crossings is part of the input. Moreover, the attained running time is polynomial for any $k \in \mathcal{O}(\log |V|)$. Using the currently fastest MAXCUT algorithm for planar graphs [LP12, SWK90], our algorithm yields a running time of $\mathcal{O}(2^k \cdot (|V| + k)^{3/2} \log(|V| + k))$.*

Quite sophisticated results by Grohe [Gro04] and Karabayashi and Reed [KR07] show that the problem to compute the crossing number of a graph is in FPT (even in linear time) with respect to its natural parameterization: Given a graph G and a number $k \in \mathbb{N}$, we can answer the question “ $\text{cr}(G) \leq k$?” in time $\mathcal{O}(f(k) \cdot n)$.

In case of a yes-instance, we obtain a corresponding crossing configuration \mathcal{X} as a witness. The computable function $f(k)$ is purely dependent on k . However, the dependency $f(k)$ is double exponential, and the algorithm far from being practical. Still, these results formally allow us to get rid of the requirement that \mathcal{X} is part of the input:

Corollary 3.2.7. *Given an edge-weighted undirected graph G . Computing a maximum cut in G is FPT with parameter $\text{cr}(G)$.*

3.3 Minor Crossing Number

We say G is a *minor* of H , denoted by $G \preceq H$, if G can be obtained from H by deletion and contraction of edges. The *minor crossing number* of G is given by $\text{mcr}(G) = \min\{\text{cr}(H) : G \preceq H\}$. A *realization* of $\text{mcr}(G)$ is a pair (H, \mathcal{X}) with $G \preceq H$ and \mathcal{X} being a crossing configuration of H with $\text{mcr}(G)$ crossings. It is easy to see that for graphs G' of maximum degree 3 we have $\text{cr}(G') = \text{mcr}(G')$. Similarly, any graph G allows a realizing graph H ($\text{cr}(H) = \text{mcr}(G)$) of maximum degree 3 where vertices of G are replaced by disjoint cubic trees.

By definition we always have $\text{mcr}(G) \leq \text{cr}(G)$; as such $\text{mcr}(G)$ can be a stronger FPT-parameter. Also, in contrast to crossing number, the minor crossing number is monotone with respect to graph minors, i.e., the family $\{G : \text{mcr}(G) \leq k\}$ is minor closed. Thus, by [RS95], we can (theoretically) check whether $\text{mcr}(G) \leq k$ in $\mathcal{O}(|V(G)|^3)$ time for fixed $k \in \mathbb{N}$.

Given a connected graph G with $\text{mcr}(G) = k$, we can obtain a graph H from G realizing $\text{mcr}(G)$ in polynomial time as follows: Choose a node v of degree at least 4. Try different pairs of neighbors $w_1, w_2 \in N(v)$ until finding the first with $\text{mcr}(\tilde{G}) \leq k$, where \tilde{G} is obtained from G by splitting v into two nodes v_1 and v_2 with $N(v_1) = \{v_2, w_1, w_2\}$, $N(v_2) = (N(v) \cup \{v_1\}) \setminus \{w_1, w_2\}$ ⁴. We call the edge v_1v_2 a *split edge*. Iterating this for each high degree node, yields a graph H of maximum degree 3 realizing $\text{mcr}(G) = \text{cr}(H)$. Note that H has at most $\mathcal{O}(|E(G)|)$ nodes.

Let $M = -3 \cdot \sum_{e \in E(G)} |c_e|$. Attaching the weight M to each split edge, we can make sure that these edges are not in any maximum cut of H . Clearly, the cuts in H not containing any split edges are in one-to-one correspondence with cuts in G . Using Theorem 3.2.5, we obtain an algorithm computing a maximum cut on G parameterized by the $\text{mcr}(G)$. Similarly to Corollary 3.2.7 we do not require an explicit realization as part of the input (using the above construction method for H).

⁴Observe that in general this splitting operation may increase mcr ; we search for a split (which has to exist) for which it does not increase. Since the split is an inverse minor operation, mcr can never decrease.

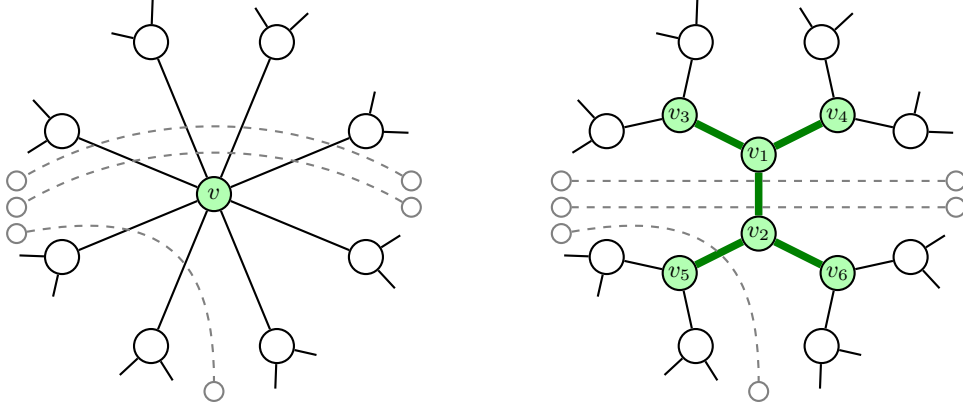


Figure 3.3: Visualization of the split operation to obtain an mcr -realization. Left: part of a graph G with $\text{cr}(G) > \text{mcr}(G)$. Right: part of \tilde{G} after splitting v five times. Bold green lines denote *split edges*.

Corollary 3.3.1. (i) Let $G = (V, E, c)$ be an edge-weighted undirected graph with $\text{mcr}(G) = k$, (H, \mathcal{X}) a realization of $\text{mcr}(G)$, and $p(n)$ be a polynomial upper bound on the running time of a MAXCUT algorithm on planar graphs with n nodes. We can compute a maximum cut in G in $\mathcal{O}(2^k \cdot p(|E(G)| + k))$ time.

(ii) Given an edge-weighted undirected graph G , computing a maximum cut in G is FPT with parameter $\text{mcr}(G)$.

3.4 Open Problems

Given a graph together with a feasible crossing configuration with k crossings, we previously only knew that MAXCUT is polynomial time solvable if k is constant *and* the graph is 1-planar, i.e., each edge is involved in at most one crossing. The runtime dependency on k has been to the order of 3^k [DKM18].

Herein, we improved on this in several ways: Firstly, we decreased the dependency on k to the order of 2^k . Secondly, we extended the applicability to *any* graph with (at most) k crossings: our parameter becomes the true crossing number of the graph, without any 1-planarity restriction. This shows that MAXCUT is in FPT with respect to the graph's crossing number. Moreover, we achieve these improvements by introducing simpler ideas than those proposed for the former result, yielding an overall surprisingly simple algorithm. Compared to the result of Kobayashi et al. [KKMT19b], we are able to stay within the realm of MAXCUT. Finally, our result naturally carries over to the minor crossing number.

The *skewness* of a graph is the minimum number of edges to remove such that the graph becomes planar. The *genus* of a graph is the minimum oriented genus of a surface onto which the graph can be embedded without crossings. In

FPT research, there are many algorithmic approaches that consider graphs with bounded genus g , see, e.g. [BFL⁺16, CKP⁺07, EFF04, FLRS11]. However, the obtained FPT algorithms are typically parameterized by the objective value z , or by the combined parameter (z, g) . There are much fewer results that obtain FPT algorithms parameterized purely with g . Notable examples are the graph genus problem itself [Moh99] (where z and g coincide by definition), and the graph isomorphism problem [Kaw15] (which generalizes the linear-time algorithm for the problem on planar graphs). There are even fewer parameterized results with respect to skewness; the probably best known example is that maximum flow can be solved in the running time of planar graphs, if the graph's skewness is fixed [HW07]. Our above algorithm seems to be the first time that the crossing number has been proposed as an efficient non-trivial FPT parameter for any widely known problem.

Besides the weight-restricted case of [GLV01] (briefly described in the introduction), it is unclear whether MAXCUT could be FPT with respect to either skewness or genus. We deem this an interesting question for further research.

Chapter 4

On the Bond Polytope

This chapter is based on [CJN20].

The (also NP-complete) maximum bond problem (MAXBOND) is obtained from MAXCUT by adding a natural connectivity requirement for both sides of the cut. Note that both problem variants can be seen as the “inverse” of the (polynomial-time solvable) minimum cut problem, where connectivity of the partition sides arises naturally.

Formally, considering a graph $G = (V, E)$ a cut $\delta(S) \subseteq E$ is a bond, if $G[S]$ and $G[V \setminus S]$ are connected. Given edge weights c_e , MAXBOND is the problem of finding a node subset $S \subseteq V$ that maximizes $\sum_{e \in \delta(S)} c_e$ under the restriction that $\delta(S)$ is a bond.

This problem is known under a variety of names including maximum minimal cut [EHKK19], largest bond [DLP⁺19], connected max cut [Cha20], and maximum connected sides cut problem [Cha17]. To avoid confusion with the maximum *one-sided* connected cut problem [DEH⁺20, EHKK19, GHK⁺18, HKM⁺15, HKM⁺20] we stick to the naming maximum bond. The research on MAXBOND is driven by applications like image segmentation [VKR08], forest planning [CCG⁺13], and computing market splittings [GKL⁺19].

MAXBOND is known to be NP-complete [HV91], even when restricted to 3-connected planar graphs [HV91] or bipartite planar graphs [DLP⁺19, DEH⁺20, EHKK19]. Conversely, MAXBOND is solvable in linear time on series-parallel graphs [Cha17]. Moreover there is an extensive study of the parameterized complexity of MAXBOND [DLP⁺19, DEH⁺20, EHKK19]. On the other hand, it is known that there is no constant factor approximation (if $P \neq NP$) [DLP⁺19, DEH⁺20].

Besides the mentioned algorithmic results there is only little knowledge on the maximum bonds in general graphs: Ding, Dziobiak and Wu proved that the maximum bond in any simple 3-connected graph G with $|V(G)| = n$ has size at

least $\frac{2}{17}\sqrt{\log n}$ and conjectured that the maximum bond in such a graph has size $\Omega(n^{\log_3 2})$ [DDW16]. This conjecture was verified by Flynn for several graph classes including planar graphs [Fly17] but remains open in general.

In this chapter we consider MAXBOND from a polyhedral viewpoint. Approaching MAXBOND by linear programming yields the *bond polytope* which is closely related to the cut polytope: The bond polytope $\text{BOND}(G)$ of G is defined as the convex hull of all incidence vectors of bonds, i.e.,

$$\text{BOND}(G) = \text{conv}(\{x^\delta : \delta \text{ is a bond in } G\}) \subseteq \mathbb{R}^E.$$

We start the structural study of bond polytopes.

Organization of this chapter. After introducing the bond polytope as the main object of this chapter, we discuss the relation of cut- and bond polytopes in Chapter 4.1. This includes the observation that several fundamental properties of cut polytopes do not carry over to bond polytopes.

In Chapter 4.2 we study how graph modifications (such as node splitting and edge contraction) effect bond polytopes and their facets.

In Chapter 4.3 we present an efficient (linear time) reduction of MAXBOND on arbitrary graphs to MAXBOND on 3-connected graphs. This algorithm can be used as an argument, why one can focus on investigating bond polytopes of 3-connected graphs.

Next, we turn our attention to edge- and cycle inequalities in bond polytopes, as they are known to be highly important in cut polytopes. In Chapter 4.4 we present *non-interleaved* cycle inequalities, a class of facet-defining inequalities arising from a special class of cycles. After this, we discuss a generalization of such inequalities as well as edge inequalities in Chapter 4.5.

We close this chapter by considering $(K_5 - e)$ -minor-free graphs in Chapter 4.6. We present a linear description of all bond polytopes of planar 3-connected such graphs. Combined with our reduction strategy from Chapter 4.3, we can complement this with a linear-time algorithm for such graphs, improving (and fixing, see below) the current quadratic-time algorithm.

A note on computing maximum bonds in $(K_5 - e)$ -minor-free graphs.

While the main focus of our work herein is to better understand the bond polytope and its facets, our results have direct algorithmic consequences—among others, on graphs with forbidden $(K_5 - e)$ -minor. Recently, an algorithm was proposed to solve MAXBOND on such graphs in quadratic time [Cha20]. The key idea is to consider the graph’s decomposition via 2-sums, and solving each component in quadratic time. However, the proposed algorithm’s description is quite rough (e.g., it does not discuss how to efficiently obtain the 2-sum decomposition to start with)

and contains a severe flaw, leading to an exponential instead of a quadratic overall running time: In [Cha20] only the case of two subgraphs, joined via a 2-sum, is discussed (either their common vertices are in the same partition side or not). One can hence compute both cases for both subgraphs and find the best choice. It is never discussed how to proceed in the case of more than two components. In fact, chaining this algorithm would yield an exponential running time of the order of $\Omega(2^c)$ for c components; $(K_5 - e)$ -minor-free graphs can have $c \in \Theta(n)$.

We resolve all these issues by giving an algorithm for the considered graph class that only requires linear running time.

4.1 First Properties and Comparison to $\text{CUT}^\square(G)$

We start this section by introducing the bond polytope. Afterwards, we start the study of bond polytopes by investigating their relation to cut polytopes. In particular, we discuss whether some fundamental results on cut polytopes carry over to bond polytopes.

We first observe that since by definition $\text{BOND}(G) \subseteq \text{CUT}^\square(G)$ for any graph G , every facet-defining inequality of $\text{CUT}^\square(G)$ is valid for $\text{BOND}(G)$. In [BM86] it was shown that 1-dimensional faces of $\text{CUT}^\square(G)$ can be characterized by bonds.

Proposition 4.1.1. [BM86, Theorem 4.1] *Let $G = (V, E)$ be a connected graph and $\delta, \gamma \subseteq E$ be cuts. Then, x^δ and x^γ are the vertices of a 1-dimensional face of $\text{CUT}^\square(G)$ if and only if their symmetric difference $\delta \Delta \gamma$ is a bond.*

As an almost immediate consequence we get an easy criterion for a vertex of $\text{CUT}^\square(G)$ being the incidence vector of a bond.

Theorem 4.1.2. *Let $G = (V, E)$ be a connected graph. Then, the following hold:*

- (i) *The vertices of $\text{BOND}(G)$ are $\mathbf{0}$ and its neighbors in $\text{CUT}^\square(G)$. In particular, $\text{cone}(\text{CUT}^\square(G)) = \text{cone}(\text{BOND}(G))$.*
- (ii) $\dim \text{BOND}(G) = |E|$.
- (iii) *A homogeneous inequality $a^\top x \leq 0$ is facet-defining for $\text{BOND}(G)$ if and only if it is facet-defining for $\text{CUT}^\square(G)$.*

Proof. Statement (i) follows directly from Proposition 4.1.1 and the fact that \emptyset is a bond. Now, (iii) is implied by (i) and (ii) follows from (i) since $\dim \text{CUT}^\square(G) = |E|$. \square

Given a graph G , an edge $e \in E(G)$, and a set $S \subseteq V(G)$, the incidence vector of $\delta_{G-e}(S)$ is obtained from the incidence vector of $\delta_G(S)$ by removing the

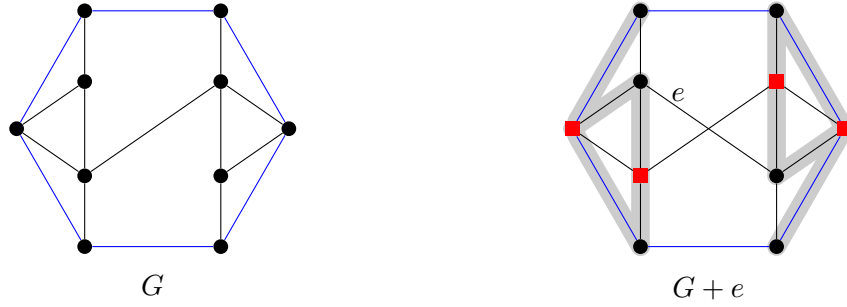


Figure 4.1: Graphs from Example 4.1.5. Marked edges in $G+e$ are those contained in the bond.

coordinate corresponding to e . As a consequence $\text{CUT}^\square(G-e)$ is the projection of $\text{CUT}^\square(G)$ onto the hyperplane $\{x_e = 0\}$. The next example shows that this does not carry over to bond polytopes.

Example 4.1.3. For any $e = vw \in E(K_4)$ the cut $\delta_{K_4}(\{v, w\})$ is a bond in K_4 . However, $\delta_{K_4-e}(\{v, w\})$ is a cut but no bond in K_4-e . In particular, $\text{BOND}(K_4-e)$ is not the projection of $\text{BOND}(K_4)$. ◀

Considering a graph G and some facet-defining inequality $a^\top x \leq b$ of $\text{CUT}^\square(G)$ it is known that $a^\top x \leq b$ is a facet of $\text{CUT}^\square(\text{supp}(a))$. This is not true in general for bond polytopes as shown in the following example.

Example 4.1.4. Considering $C_6 \subseteq K_{3,3}$, the inequality $\sum_{e \in E(C_6)} x_e \leq 4$ is not even tight for $\text{BOND}(C_6)$ but facet-defining for $\text{BOND}(K_{3,3})$. Indeed, this example generalizes to $C_{2n} \subseteq V_n$ for arbitrary $n \geq 3$ (see Chapter 4.5 for the definition of the Wagner graph V_n). ◀

Conversely, it is well-known that if $H \subseteq G$ is a subgraph, the 0-lifting, i.e., the lifting by taking the induced inequality in $\mathbb{R}^{E(G)}$, of each valid inequality of $\text{CUT}^\square(H)$ is valid for $\text{CUT}^\square(G)$. Again, this also does not carry over to bond polytopes.

Example 4.1.5. Consider the graphs in Figure 4.1 and denote the outer (blue) cycle by C . Then, $\sum_{e \in E(C)} x_e \leq 2$ defines a facet of $\text{BOND}(G)$, but this inequality is not even valid for $\text{BOND}(G+e)$, as the (red) square nodes induce a bond δ with $|\delta \cap E(C)| > 2$. ◀

In contrast to this, contracting an edge e corresponds to intersecting the bond polytope with the hyperplane $\{x_e = 0\}$ as it is the case for cut polytopes.

Observation 4.1.6. Let G be a graph and $e \in E(G)$. Then, $\text{BOND}(G/e) = \text{BOND}(G) \cap \{x_e = 0\}$.

The most prominent symmetries of cut polytopes are given by graph automorphisms and switchings (see Lemma 1.5.2). While graph automorphisms clearly give rise to symmetries of bond polytopes, switching does not in general:

Observation 4.1.7. *Let $G = (V, E)$ be a graph, $W \subseteq V(G)$, and $a^\top x \leq b$ be facet-defining for $\text{BOND}(G)$. We consider the following situations:*

- (i) *If $b = 0$ and $\delta(W)$ is a bond satisfying $a^\top x^{\delta(W)} = 0$, switching $a^\top x \leq b$ at W gives a facet of $\text{BOND}(G)$.*
- (ii) *If $b = 0$ and $\delta(W)$ is a bond with $a^\top x^\delta < 0$, the inequality obtained by switching $a^\top x \leq 0$ at W is not facet-defining for $\text{BOND}(G)$ in general.*
- (iii) *If $b \neq 0$, switching $a^\top x \leq b$ at a node set W does not define a facet of $\text{BOND}(G)$ in general. It might not even be valid for $\text{BOND}(G)$ (even if $\delta(W)$ is a bond).*

Proof. In statement (i) we consider the switching of a homogeneous facet of $\text{CUT}^\square(G)$ at a solution of itself. This switching yields a homogeneous facet of $\text{CUT}^\square(G)$ and thus a facet of $\text{BOND}(G)$.

Statement (ii) and (iii) can be shown via examples; we consider certain facets of $\text{BOND}(C_n)$. The facet description of $\text{BOND}(C_n)$ is discussed in Theorem 4.4.8. For statement (ii) consider the facet-defining inequality

$$x_e - \sum_{\substack{f \in E(C_n) \\ f \neq e}} x_f \leq 0.$$

for some $e \in E(C_n)$. Switching this at $\{v\}$ for some $v \in V(C_n)$ that is not incident to e we obtain the inequality

$$x_e + \sum_{f \in \delta(\{v\})} x_f - \sum_{\substack{f \in E(C_n) \setminus \delta(\{v\}) \\ f \neq e}} x_f \leq 2.$$

It follows directly from Theorem 4.4.8 that the latter does not define a facet.

For statement (iii) consider a cycle C_n with $n \geq 4$ and the facet-defining inequality $\sum_{e \in E(C)} x_e \leq 2$ for $\text{BOND}(C_n)$. Switching at an arbitrary node $v \in V(C_n)$ gives the inequality

$$\sum_{\substack{e \in E(C_n) \\ v \notin e}} x_e - \sum_{\substack{e \in E(C_n) \\ v \in e}} x_e \leq 0.$$

But this is violated by $x^{\delta(\{w\})}$ for each $w \in V(C_n)$ that is not adjacent to v . \square

4.2 Constructing Facets from Facets

In Chapter 1.4 we recapitulated the extensive study considering the effect of graph operations (such as node splitting, edge subdivisions, edge contraction, and deletion of edges) on cut polytopes and their facet-defining inequalities. Motivated by this, we start an investigation of the effect of such graph operations on bond polytopes and their facets.

Theorem 4.2.1 (Node splitting). *Let $G = (V, E)$ be a connected graph, $v \in V$, and $a^\top x \leq b$ be facet-defining for $\text{BOND}(G)$. Obtain $\bar{G} = (\bar{V}, \bar{E})$ as follows: replace v by two adjacent nodes v_1 and v_2 and distribute the edges incident to v arbitrarily among v_1 and v_2 . Set $\varphi: \bar{E} \setminus \{v_1v_2\} \rightarrow E$ by*

$$\varphi(e) = \begin{cases} e, & \text{if } v_1, v_2 \notin e, \\ v_i, & \text{if } e = v_iw \ (i = 1, 2). \end{cases}$$

Let ω be the value of a maximum bond in $\bar{G} - v_1v_2$ separating v_1 and v_2 with respect to the edge weights given by $a_{\varphi(e)}$. Now, set \bar{a} by

$$\bar{a}_e = \begin{cases} a_{\varphi(e)}, & \text{if } e \neq v_1v_2, \\ b - \omega, & \text{if } e = v_1v_2. \end{cases}$$

Then, $\bar{a}^\top x \leq b$ defines a facet of $\text{BOND}(\bar{G})$.

Proof. First, we show that $\bar{a}^\top x \leq b$ is a valid inequality for $\text{BOND}(\bar{G})$. Each bond in \bar{G} not containing v_1v_2 corresponds to a bond in G . Hence, it is easy to see that all such bonds satisfy the inequality under consideration. Now, let $\delta \subseteq \bar{E}$ be a bond with $v_1v_2 \in \delta$. Then

$$\bar{a}^\top x^\delta = \bar{a}_{v_1v_2} x_{v_1v_2}^\delta + \sum_{\substack{e \in \bar{E} \\ e \neq v_1v_2}} \bar{a}_e x_e^\delta = (b - \omega) x_{v_1v_2}^\delta + \sum_{e \in E} a_e x_e^{\varphi(\delta)} \leq (b - \omega) + \omega = b.$$

It remains to show that $\bar{a}^\top x \leq b$ is indeed facet-defining. Let $m = |E|$. Since $a^\top x \leq b$ defines a facet of $\text{BOND}(G)$, there exist $W_1, \dots, W_m \subseteq V$ with $v \notin W_i$ such that $x^{\delta_G(W_i)}$ satisfies $a^\top x = b$ for each $i \in [m]$ and $x^{\delta_G(W_1)}, \dots, x^{\delta_G(W_m)}$ are affinely independent. It is easy to see that $\delta_i = \delta_{\bar{G}}(W_i) \subseteq \bar{E}$ is a bond (in \bar{G}) with $v_1v_2 \notin \delta_i$ satisfying $\bar{a}^\top x = b$.

Now let $W_0 \subseteq \bar{V}$ such that $\delta_{\bar{G}-v_1v_2}(W_0)$ is a bond in $\bar{G} - v_1v_2$ separating v_1 and v_2 with $\sum_{e \in \delta_{\bar{G}-v_1v_2}(W_0)} a_e = \omega$. Hence, for $\delta_0 = \delta_{\bar{G}}(W_0) \subseteq \bar{E}$ we have $\bar{a}^\top x^{\delta_0} = \omega + a_{v_1v_2} = \omega + b - \omega = b$. Since $v_1v_2 \in \delta_0$ and $v_1v_2 \notin \delta_i$ for $1 \leq i \leq m$, it is easy to see that $x^{\delta_0}, x^{\delta_1}, \dots, x^{\delta_m}$ are affinely independent. \square

Theorem 4.2.2 (Replacing a node by a triangle). *Let $G = (V, E)$ be a connected graph, $v \in V$, and $a^\top x \leq b$ be facet-defining for $\text{BOND}(G)$. Obtain $\bar{G} = (\bar{V}, \bar{E})$ from G by replacing v by a triangle on vertices $\{v_1, v_2, v_3\}$ and distributing the edges incident to v arbitrarily among v_1, v_2 , and v_3 .*

Set $\varphi: \bar{E} \setminus \{v_1v_2, v_1v_3, v_2v_3\} \rightarrow E$ by

$$\varphi(e) = \begin{cases} e, & \text{if } v_1, v_2, v_3 \notin e, \\ vw, & \text{if } e = v_iw \ (i \in [3]). \end{cases}$$

For $i = 1, 2, 3$, let $S_i \subseteq \bar{V}$ such that $S_i \cap \{v_1, v_2, v_3\} = \{v_i\}$ and $\delta_{\bar{G}}(S_i)$ is a maximum bond with respect to edge weight 0 attached to v_jv_k ($j, k \in [3]$) and $a_{\varphi(e)}$ for each other edge e . Denote by ω_i the weight of $\delta_{\bar{G}}(S_i)$ with respect to these weights. Define $\bar{a} \in \mathbb{R}^{\bar{E}}$ by

$$\bar{a}_e = \begin{cases} a_{\varphi(e)}, & \text{if } e \neq v_iv_j \ (i, j \in [3]), \\ \frac{1}{2}(b - \omega_1 - \omega_2 + \omega_3), & \text{if } e = v_1v_2, \\ \frac{1}{2}(b - \omega_1 + \omega_2 - \omega_3), & \text{if } e = v_1v_3, \\ \frac{1}{2}(b + \omega_1 - \omega_2 - \omega_3), & \text{if } e = v_2v_3. \end{cases}$$

Then, $\bar{a}^\top x \leq b$ defines a facet of $\text{BOND}(\bar{G})$.

Proof. The proof of this theorem is analogous to the one of Theorem 4.2.1. We only have to utilize that $x^{\delta_G(S_1)}, x^{\delta_G(S_2)}, x^{\delta_G(S_3)}$ all satisfy $\bar{a}^\top x = b$ and are affinely independent to all bond-vectors $x^{\delta_{\bar{G}}(S)}$ with $S \subseteq \bar{V}$. \square

Lemma 4.2.3. *Let $G = (V, E)$ be a connected graph, $e \in E$, and $a^\top x \leq b$ be facet-defining for $\text{BOND}(G)$. Assume that $\{a^\top x \leq b\} \neq \{-x_e \leq 0\}$ and $a_e \neq 0$. Then, there exists some bond δ such that $e \in \delta$ and $a^\top x^\delta = b$.*

Proof. Assume there is no such δ and consider the inequality $a^\top x - \lambda x_e \leq b$ for some $\lambda > 0$. Validity follows since $a^\top x - \lambda x_e \leq a^\top x$ for each $x \in [0, 1]^{|E|}$. Moreover, by assumption each bond satisfying $a^\top x = b$ satisfies $a^\top x - \lambda x_e = b$. Hence, both inequalities describe the same face of $\text{BOND}(G)$ contradicting $a^\top x \leq b$ being facet-defining. \square

Theorem 4.2.4 (Subdividing an edge). *Let $G = (V, E)$ be a connected graph and $a^\top x \leq b$ be facet-defining for $\text{BOND}(G)$. Let $e = vw \in E$ with $a_e \leq \frac{b}{2}$ and $\{a^\top x \leq b\} \neq \{-x_e \leq 0\}$.*

Obtain $\bar{G} = (\bar{V}, \bar{E})$ by splitting e into $e_1 = v\bar{v}$ and $e_2 = \bar{v}w$ (for a new node \bar{v}). For $f \in \bar{E}$ set

$$\bar{a}_f = \begin{cases} a_f, & \text{if } f \in E, \\ a_e, & \text{if } f \in \{e_1, e_2\}. \end{cases}$$

Then, $\bar{a}^\top x \leq b$ is facet-defining for $\text{BOND}(\bar{G})$.

Proof. For each cut $\delta \subseteq \bar{E}$ in \bar{G} we have $|\delta \cap \{e_1, e_2\}| \leq 1$ or $\delta = \delta_{\bar{G}}(\bar{v})$. Hence, it is easy to see that $\bar{a}^\top x \leq b$ is valid for $\text{BOND}(\bar{G})$.

Since $a^\top x \leq b$ is facet-defining for $\text{BOND}(G)$ there exist $U_1, \dots, U_m \subseteq V$ ($m = |E|$) such that each $\delta_G(U_i)$ is a bond in G satisfying $a^\top x^{\delta(U_i)} = b$. We may assume $v \in U_i$ for all $i \in [m]$. Setting $\bar{U}_i = U_i \cup \{\bar{v}\}$ it is easy to see that $\delta_i = \delta_{\bar{G}}(\bar{U}_i)$ is a bond satisfying $\bar{a}^\top x^{\delta_i} = b$. Note that $e_1 \notin \delta_i$ for each $i \in [m]$. By Lemma 4.2.3 there exists $U_0 \subseteq V$ such that $v \in U_0$, $w \notin U_0$, and $a^\top x^{\delta_G(U_0)} = b$. We conclude that $e_1 \in \delta_{\bar{G}}(U_0)$ and thus, $x^{\delta_{\bar{G}}(U_0)}, x^{\delta_1}, \dots, x^{\delta_m}$ are affinely independent. Hence, $\bar{a}^\top x \leq b$ is facet-defining for $\text{BOND}(\bar{G})$. \square

Iteratively applying this theorem yields the following corollary:

Corollary 4.2.5 (Replacing an edge by a path). *Let $G = (V, E)$ be a connected graph and $a^\top x \leq b$ be facet-defining for $\text{BOND}(G)$. Let $e = vw \in E$ and assume that $a_e \leq \frac{b}{2}$ and $\{a^\top x \leq b\} \neq \{-x_e \leq 0\}$.*

Obtain $\bar{G} = (\bar{V}, \bar{E})$ by subdividing e into edges e_1, \dots, e_k for arbitrary $k \in \mathbb{N}$, $k \geq 2$. For $f \in \bar{E}$ set

$$\bar{a}_f = \begin{cases} a_f, & \text{if } f \in E, \\ a_e, & \text{if } f \in \{e_1, \dots, e_k\}. \end{cases}$$

Then, $\bar{a}^\top x \leq b$ is facet-defining for $\text{BOND}(\bar{G})$.

Next, we consider the inverse, i.e., the replacement of an induced path by an edge. To this end, we start by investigating coefficients of facet-defining inequalities on edges contained in induced paths.

Lemma 4.2.6. *Let $G = (V, E)$ be a connected graph, $P \subseteq G$ be an induced path, $a^\top x \leq b$ be facet-defining for $\text{BOND}(G)$, and \mathcal{E} be the set of all bonds δ in G satisfying the equality $a^\top x^\delta = b$. Assume that $E(P) \cap \text{supp}(a) \neq \emptyset$ and $\{a^\top x \leq b\} \neq \{-x_e \leq 0\}$ for each $e \in E(P)$. Then:*

- (i) *If $|\delta \cap E(P)| \leq 1$ for all $\delta \in \mathcal{E}$, we have $a_e = a_f$ for all $e, f \in E(P)$.*
- (ii) *If there exists some $\delta^* \in \mathcal{E}$ with $|\delta^* \cap E(P)| = 2$, we have either $a_e = a_f$ for all $e, f \in E(P)$ or there is a unique $e^* \in E(P)$ such that $a_{e^*} > a_e$ and $a_e = a_f$ for all $e, f \in E(P) \setminus \{e^*\}$.*

In particular, we have $a_e \neq 0$ for some $e \in E(G) \setminus E(P)$ or $\max_{e \in E(P)} a_e > 0$.

Proof. By [Zie00, Theorem 5] we may assume $a \in \mathbb{Z}^E$ and since $\mathbf{0} \in \text{BOND}(G)$ we have $b \in \mathbb{Z}_{\geq 0}$. Let $M = \max_{e \in E(P)} \{a_e\}$ and $e^* \in E(P)$ with $a_{e^*} = M$. Set

$N = \max_{e \in E(P) \setminus \{e^*\}} \{a_e\}$ and define $c \in \mathbb{R}^E$ by

$$c_e = \begin{cases} a_e, & \text{if } e \notin E(P), \\ M, & \text{if } e = e^*, \\ N, & \text{else.} \end{cases}$$

Note that we might have $M = N$. Let δ be an arbitrary bond in G . If $\delta \cap E(P) = \emptyset$, we have $a^\top x^\delta = c^\top x^\delta \leq b$. If $|\delta \cap E(P)| = \{f\}$ for some $f \in E(P)$, we consider the bond $\delta' = (\delta \setminus \{f\}) \cup \{e^*\}$ and obtain $c^\top x^\delta = a^\top x^{\delta'} \leq b$. If $|\delta \cap E(P)| = 2$, we consider the bond $\delta' = (\delta \setminus (\delta \cap E(P))) \cup \{e^*, f^*\}$ for some $f^* \in E(P)$ with $a_{f^*} = N$ and obtain $c^\top x^\delta = a^\top x^{\delta'} \leq b$. Hence, $c^\top x \leq b$ is valid for $\text{BOND}(G)$. Since $a^\top x \leq b$ can be obtained from $c^\top x \leq b$ by adding homogeneous edge inequalities, we have either $c = a$ or $c = \mathbf{0}$. The latter case implies that $a^\top x \leq b$ is a homogeneous edge inequality contradicting the assumption.

Thus, in the following we can assume that $a = c$. Then, statement (ii) follows from the previous discussion. To prove statement (i) assume, that $|\delta \cap E(P)| \leq 1$ for all $\delta \in \mathcal{E}$ and assume that $m < M$. Defining $d \in \mathbb{R}^E$ by

$$d_e = \begin{cases} a_e, & \text{if } e \notin E(P), \\ M, & \text{if } e = e^*, \\ m + 1, & \text{else,} \end{cases}$$

we show that $a^\top x \leq b$ cannot be facet-defining for $\text{BOND}(G)$. Similar as above, we have $d^\top x^\delta \leq b$ for all bonds δ with $|\delta \cap E(P)| \leq 1$.

Since P is an induced path, a bond in G contains at most 2 edges from P . Hence, it remains to prove validity for all bonds picking 2 edges from P . To this end, let δ be such a bond. By assumption we have $a^\top x^\delta < b$ and thus, since both values are integers it follows that $b - 1 \geq a^\top x^\delta = M + N$. Hence, $d^\top x^\delta = M + N + 1 \leq b$. Since $d = \mathbf{0}$ would imply that $a^\top x \leq b$ is a homogeneous edge inequality, $d^\top x \leq b$ defines a proper face of $\text{BOND}(G)$. Thus, $d^\top x \leq b$ dominates $a^\top x \leq b$ contradicting the assumption that the latter inequality is facet-defining.

The ‘‘in particular’’-part holds since otherwise $a^\top x \leq b$ is dominated by homogeneous edge inequalities. \square

Example 4.2.7. Indeed, all cases of the previous lemma can occur. We may see this considering facet-defining inequalities of $\text{BOND}(C_n)$, which are formally discussed later in Theorem 4.4.8. Let $e \in E(C_n)$, and consider the facet-defining

inequalities

$$\sum_{e \in E(C_n)} x_e \leq 2 \quad \text{and} \quad (4.1)$$

$$x_e - \sum_{\substack{f \in E(C_n) \\ f \neq e}} x_f \leq 0. \quad (4.2)$$

An example for statement (i) is given by inequality (4.2) and an arbitrary induced path P not containing e . For the first case of statement (ii) consider inequality (4.1) and an arbitrary induced path $P \subseteq C_n$; for the second case consider inequality (4.2) and an induced path $P \subseteq C_n$ with $e \in E(P)$. ◀

Given the previous lemma, we are now prepared to investigate, how replacing a path by an edge effects facet-defining inequalities of bond polytopes.

Theorem 4.2.8 (Replacing a path by an edge). *Let $G = (V, E)$ be a connected graph, $a^\top x \leq b$ be facet-defining for $\text{BOND}(G)$, and P be an induced path in G . Denote by \mathcal{E} the set of all bonds δ in G satisfying the equality $a^\top x^\delta = b$. Assume that $\{a^\top x \leq b\} \neq \{-x_e \leq 0\}$ for each $e \in E(P)$ and either there exist $e, f \in E(P)$ with $a_e \neq a_f$ or there exists no bond $\delta \in \mathcal{E}$ with $|\delta \cap E(P)| = 2$.*

Let $\bar{G} = (\bar{V}, \bar{E})$ be obtained from G by replacing P by a single edge p . Set $M = \max_{e \in E(P)} \{a_e\}$ and define $\bar{a} \in \mathbb{R}^{\bar{E}}$ by

$$\bar{a}_e = \begin{cases} a_e, & \text{if } e \in E, \\ M, & \text{if } e=p. \end{cases}$$

Then, $\bar{a}^\top x \leq b$ defines a facet of $\text{BOND}(\bar{G})$.

Proof. Let $\pi: \mathbb{R}^E \rightarrow \mathbb{R}^{\bar{E}}$ be the projection given by

$$\pi(x)_e = \begin{cases} x_e, & \text{if } e \neq p, \\ \sum_{e \in E(P)} x_e, & \text{if } e = p. \end{cases}$$

If there is no $\delta^* \in \mathcal{E}$ with $|\delta^* \cap E(P)| = 2$, it is straight forward to see that for each bond $\delta \in \mathcal{E}$ we have $\pi(x^\delta) \in \text{BOND}(\bar{G})$ and $\bar{a}^\top \pi(x^\delta) = b$. Thus,

$$\begin{aligned} \dim(\{\bar{a}^\top x = b\} \cap \text{BOND}(\bar{G})) &\geq \dim(\{a^\top x = b\} \cap \text{BOND}(G)) - (|E(P)| - 1) \\ &= |E| - 1 - (|E(P)| - 1) = |\bar{E}| - 1. \end{aligned}$$

Since $a^\top x \leq b$ was facet-defining, we have $\bar{a} \neq \mathbf{0}$. Moreover, since $\text{BOND}(\bar{G})$ has dimension $|\bar{E}|$, we have $\{\bar{a}^\top x = b\} \cap \text{BOND}(\bar{G}) \neq \text{BOND}(\bar{G})$ yielding that $\bar{a}^\top x \leq b$ is facet-defining.

Now, assume there exists a bond $\delta^* \in \mathcal{E}$ with $|\delta^* \cap E(P)| = 2$ and edges $e, f \in E(P)$ with $a_e \neq a_f$. By Lemma 4.2.6, there exists a unique edge $e^* \in E(P)$ with $a_{e^*} > a_f$ for all $f \in E(P) \setminus \{e^*\}$ and we have $a_e = a_f$ for all $e, f \in E(P) \setminus \{e^*\}$. Note that for each bond $\delta \in \mathcal{E}$ with $\delta \cap E(P) \neq \emptyset$ we have $e^* \in \delta$. Thus, there are exactly $|E(P)| - 1$ such bonds with $|\delta \cap E(P)| = 2$. Since $\delta \cap E(P) = \{e^*\}$ for all bonds with $|\delta \cap E(P)| = 1$, applying π on all bonds δ with $a^\top x^\delta = b$ and $|\delta \cap E(P)| \leq 1$ yields a $(|E| - 1)$ -dimensional face of $\text{BOND}(\overline{G})$. Thus, $\overline{a}^\top x \leq b$ is facet-defining for $\text{BOND}(\overline{G})$. \square

We close this section by a discussion of a bond polytope version of [BM86, Lemma 2.5]:

Lemma 4.2.9. *Let $G = (V, E)$ be a connected graph and $a^\top x \leq b$ be a valid inequality for $\text{BOND}(G)$. Moreover, let $pq \in E$ and let $S \subsetneq V \setminus \{p, q\}$.*

If $\delta(S)$, $\delta(S \cup \{p\})$, $\delta(S \cup \{q\})$, and $\delta(S \cup \{p, q\})$ are bonds satisfying $a^\top x = b$, then $a_{pq} = 0$.

Proof. One can re-use the proof of [BM86, Lemma 2.5] by simply replacing “cut” with “bond”. \square

The cut version [BM86, Lemma 2.5] of the above lemma turns out to be a powerful tool when applied to cut versions of the other above lemmata and theorems in this section: It allows the addition of further edges to the graph while retaining the facet-defining properties of the inequalities under consideration. In the context of bond polytopes, however, there often may not exist a set S yielding the required bonds (in contrast to cuts). Nonetheless, although not as versatile as its cut version, the lemma will still be crucial in later proofs.

4.3 Reduction to 3-connectivity

We show that MAXBOND can be reduced in linear time to 3-connected graphs. While a (similar) reduction was already proposed in [Cha20], it contained a gap, leading to an exponential running time. Both our reduction and the one in [Cha20] is based on the following observation. For completeness we give a full proof.

Observation 4.3.1 (Bonds over clique sums). *Let $G = G_1 \oplus_k G_2$ with $k \in [2]$ and $\delta_G \subseteq E(G)$ be a bond. Let $\delta_i = \delta_G \cap E(G_i)$ for $i \in [2]$.*

(i) *If $k = 1$, then $\delta_i = \delta_G$ and $\delta_{3-i} = \emptyset$ for either $i = 1$ or 2 .*

(ii) *If $k = 2$, let e be the unique edge in $E(G_1) \cap E(G_2)$. Either $e \in \delta(G)$ and each δ_i is a bond in G_i or $e \notin \delta_G$ and $\delta_G = \delta_i$ for either $i = 1$ or 2 .*

Proof. We prove statement (ii). It is then straight forward to verify statement (i). Let $V(G_1) \cap V(G_2) = \{v_1, v_2\}$, $e = v_1v_2$, and $S \subseteq V(G)$ such that $\delta_G = \delta_G(S)$. Setting $S_i = S \cap V(G_i)$ and $\delta_i = \delta_{G_i}(S_i)$ for $i = 1, 2$ we have $\delta_G = \delta_1 \cup \delta_2$. Clearly, δ_1 (resp. δ_2) is a bond in G_1 (resp. G_2) and we have $v_1v_2 \in \delta_G$ if and only if $v_1v_2 \in \delta_1$ and $v_1v_2 \in \delta_2$. It only remains to show that $v_1v_2 \notin \delta_G$ implies $\delta_1 = \emptyset$ or $\delta_2 = \emptyset$. If $v_1v_2 \notin \delta$, we may assume $v_1, v_2 \in S$. Since $G - v_1v_2 = (G_1 - v_1v_2) \cup (G_2 - v_1v_2)$ there cannot be $w_1, w_2 \in V(G) \setminus S$ with $w_i \in V(G_i)$ because $G - S$ would be disconnected. Thus, $V(G_i) \subseteq S$ and δ_G is a bond in G_{3-i} for either $i = 1$ or 2 . \square

In order to use this observation algorithmically, utilize the 3-connectivity decomposition of a graph.

Recall from Chapter 2.1 that each (not necessarily simple) graph $G = (V, E)$ admits a unique 3-connectivity decomposition into components, the so-called skeletons, which can be partitioned into the following sets: a set S of cycles, a set P of edge bundles (two nodes joined by at least 3 edges), and a set R of simple 3-connected graphs. Again, we use SPR-trees as a data structure considering this decomposition.

Theorem 4.3.2. *MAXBOND on arbitrary graphs can be solved in the same time complexity as on (simple) 3-connected graphs.*

Proof. Let G be an arbitrary graph and c_e denote the edge weight of $e \in E(G)$. We denote by $\omega(G)$ the weight of a maximum bond in G . Moreover, let $p(m) \in \Omega(m)$ be the running time of MAXBOND on 3-connected graphs with m edges.

If G is not 2-connected, we can find a decomposition $G = G_1 \oplus_1 \cdots \oplus_1 G_k$ into 2-connected graphs G_1, \dots, G_k in linear time with simple depth-first search. We can consider these components individually since $\omega(G) = \max_{i \in [k]} \omega(G_i)$ (see Observation 4.3.1). Thus we may assume G to be 2-connected in the following.

For an edge $e \in E(G)$, we can compute a maximum bond not containing e (resp. containing e) in the same time as $\omega(G)$ by contracting e (resp. setting its weight to a large enough value, e.g., $2 \sum_{f \in E(G) \setminus \{e\}} |c_f|$).

First, we compute the SPR-tree $T = T(G)$ of G , attach weight 0 to each virtual edge and root T at an arbitrary node $\varrho \in V(T)$. Our algorithm will iteratively prune leaves in T .

Let α be a leaf of T , β be its parent, and denote by e the common virtual edge in H_α and H_β . We compute the value ω_α^+ of a maximum bond in H_α containing e and the value ω_α^- of a maximum bond not containing e . Then, we set the weight of e in H_β to ω_α^+ , consider e as an original (no longer virtual) edge in H_β , remove α from T , and proceed with the next leaf until only ϱ remains. In the latter case, we compute ω_ϱ as the maximum bond in the skeleton H_ϱ (where all virtual edges are already transformed into original edges with some weight computed in the previous steps).

Consider the above setting when considering a leaf α , and let δ^* be a maximum bond in G . In case of $\delta^* \subseteq E(H_\alpha) \setminus \{e\}$, we have $\omega(G) = \omega_\alpha^-$. Otherwise, let G' be the graph obtained from T after the removal of α (in particular, e is considered an original edge in G' of weight ω_α^+). Then, we have:

$$\begin{aligned} \omega(G) &= \sum_{f \in \delta^*} c_f = \sum_{\substack{f \in \delta^* \cap E(G') \\ f \neq e}} c_f + \sum_{\substack{f \in \delta^* \cap E(H_\alpha) \\ f \neq e}} c_f \\ &= \sum_{\substack{f \in \delta^* \cap E(G') \\ f \neq e}} c_f + \begin{cases} \omega_\alpha^+, & \text{if } e \in \delta^* \\ 0, & \text{if } e \notin \delta^* \end{cases} \\ &= \omega(G'). \end{aligned}$$

Overall, we have $\omega(G) = \max\{\omega_\alpha^-, \omega(G')\}$. Iterating our pruning strategy, we obtain

$$\omega(G) = \max \left\{ \omega_\varrho, \max_{\alpha \in V(T(G)) \setminus \varrho} \{\omega_\alpha^-\} \right\}.$$

Observe that $T(G)$ is the original SPR-tree and the ω -values are the maximum bonds as computed by the algorithm.

Note that computing MAXBOND on P - and S -nodes can trivially be done in linear time: a maximum bond in P -nodes either picks all edges or no edge; a maximum bond in S -nodes picks either two edges of heaviest weight or none, if the sum of any two edge weights is negative. By [HT73, Lemma 15], the SPR-tree can be built in linear time and the computations in a skeleton on m' edges can be done in time $\mathcal{O}(p(m'))$. We attain an overall running time of $\mathcal{O}\left(\sum_{\alpha \in V(T)} p(|E(H_\alpha)|)\right) \leq \mathcal{O}(p(|E(G)|))$. \square

Although not necessary in the above proof, we may mention that the bond polytope corresponding to a P -node is essentially that of a single edge; the bond polytope corresponding to an S -node, i.e., of a simple cycle, is discussed in Theorem 4.4.8 below.

4.4 Non-Interleaved Cycle Inequalities

We will now start the investigation of inequalities associated to cycles. To this end, we introduce non-interleaved cycles and show that they give rise to facet-defining inequalities for bond polytopes of 3-connected graphs. We close this section by discussing this inequalities for 2-connected graphs.

Definition 4.4.1. Let G be a graph and $C \subseteq G$ be a cycle. C is *interleaved*, if there are (not necessarily neighboring) nodes $v_1, v_2, v_3, v_4 \in V(C)$ occurring along

C in this order such that there are node-disjoint paths in $G - E(C)$ connecting v_1 with v_3 and v_2 with v_4 respectively. Otherwise, C is *non-interleaved*.

Given an interleaved cycle C we call two paths P, Q witnessing the interleavedness of C *interleaving* (with respect to C). Two such paths can be found in polynomial time if they exist [RS95]. The following lemma introduces valid inequalities for bond polytopes, which we call *non-interleaved cycle inequalities*.

Lemma 4.4.2. *Let G be a connected graph and $C \subseteq G$ be a cycle. The inequality $\sum_{e \in E(C)} x_e \leq 2$ is valid for $\text{BOND}(G)$ if and only if C is non-interleaved.*

Proof. First assume C is interleaved with interleaving paths P and Q . Set

$$S = V(P) \cup \{v \in V(G) : v \text{ is disconnected from } Q \text{ in } G \setminus P\}.$$

Then, $\delta(S)$ is a bond with $|\delta(S) \cap E(C)| = 4$. Hence, $\sum_{e \in E(C)} x_e \leq 2$ is not valid.

On the other hand, if $\sum_{e \in E(C)} x_e \leq 2$ is violated there exists a bond $\delta(S')$ with $|\delta(S') \cap C| > 2$. Since there always has to be an even number of cut edges on a cycle, we must have $|\delta(S') \cap C| \geq 4$. Let P_1, \dots, P_ℓ be the components of $C - \delta(S')$ listed in their order of appearance along C . We may assume $P_i \subseteq G[S']$ for odd i . Then, there needs to exist a path $Q_1 \subseteq G[S']$ connecting P_1 and P_3 and a path $Q_2 \subseteq G - S'$ connecting P_2 and P_4 . Clearly, Q_1 and Q_2 are interleaving with respect to C . \square

Lemma 4.4.3. *Let G be 3-connected and $C \subseteq G$ be a non-interleaved cycle. Then C is an induced cycle.*

Proof. Let G be 3-connected and assume there is a non-interleaved cycle C with a chord e . Let C_1 and C_2 be the cycles such that $E(C_1) \cap E(C_2) = \{e\}$ and $C = C_1 \triangle C_2$. There is at least one node in $C_1 \setminus C_2$ and at least one node in $C_2 \setminus C_1$. Furthermore, since G is 3-connected there exists a path P with $P \cap E(C) = \emptyset$ connecting these two nodes. But then P and e are interleaving paths with respect to C . \square

The following graph-theoretic lemma is the crucial ingredient for the facet theorem shown thereafter.

Lemma 4.4.4. *Let $G = (V, E)$ be 3-connected, $C \subseteq G$ be a cycle, and $f = pq \in E \setminus E(C)$. Then, there exists an $S \subseteq V$ such that $\delta(S)$, $\delta(S \cup \{p\})$, $\delta(S \cup \{q\})$, and $\delta(S \cup \{p, q\})$ are bonds each containing two edges of C .*

Proof. We consider two cases depending on whether pq is adjacent to C or not.

Case 1: $|\{p, q\} \cap V(C)| = 1$. We may assume $p \in V(C)$ and $q \notin V(C)$. Since G is 3-connected, there exist internally node-disjoint paths P and Q with $f \notin P, Q$

connecting p and q . We may assume $|E(P) \cap E(C)| \geq 1$, since p is not a cut-node. Now, set

$$S^+ = V(P) \cup \{v \in V : v \text{ is disconnected from } C \setminus P \text{ in } G \setminus P\}$$

and $S = S^+ \setminus \{p, q\}$. In the following, we show that the cuts $\delta(S)$, $\delta(S \cup \{p\})$, $\delta(S \cup \{q\})$, and $\delta(S^+)$ are indeed bonds. To this end it suffices to prove that $G[S]$ and $G - S^+$ are both connected, and p and q are adjacent to both of these graphs.

By construction, $G - S^+$ is connected, and the nodes p and q are adjacent to $G[S]$. Moreover, p is incident to two edges in C and only one of these is contained in $G[S^+]$. Thus, p is adjacent to $G - S^+$. Furthermore, the path Q connects q and C , yielding that q is adjacent to $G - S^+$. For each node $v \in S$ there are three disjoint paths P'_1 , P'_2 , and P'_3 connecting v and C . Since $v \in S$ we have $V(P'_i) \cap V(P) \neq \emptyset$ for each $i \in [3]$. Since at most two of these paths contain p or q , there is a path connecting v with some node in $V(P) \setminus \{p, q\}$. Hence, $G[S]$ is connected.

Case 2: $p, q \notin V(C)$. We prove that there are paths P_1, P_2 connecting p with C , and Q_1, Q_2 connecting q with C such that the paths of the triplets (P_1, P_2, Q_1) and (P_1, Q_1, Q_2) are pairwise disjoint within their triplet. Then, we may assume (possibly after exchanging indices) that there is a path $R \subseteq C$ connecting P_1 and Q_1 such that $R \cap P_2 = \emptyset$ and $R \cap Q_2 = \emptyset$. Then, setting $S' = V(R) \cup V(P_1) \cup V(Q_1)$ it is straight forward to verify that

$$S^+ = S' \cup \{v \in V(G) : v \text{ is disconnected from } C \setminus R \text{ in } G - S'\}$$

and $S = S^+ \setminus \{p, q\}$ yield the claimed bonds (cf. Figure 4.2).

We start by proving the existence of a triplet (P_1, P_2, Q) of disjoint paths P_1, P_2 connecting p with C and Q connecting q with C . For a path P and nodes $v, w \in V(P)$ we denote by $P[v:w]$ the subpath of P from v to w . Since G is 3-connected, by Menger's theorem, there exist three disjoint (except of p) paths P_1, P_2, P_3 connecting p with C . If one of them, say P_3 , contains q we are done by choosing $Q = P_3 \setminus (P_3[p:q] - q)$. Otherwise, since G is 3-connected, there exists a path Q_0 connecting q and C . If Q_0 is not the claimed path Q , it intersects at least one of the P_i . We may assume that the first such intersection is between Q_0 and P_3 and is at a node x . Then, we set $Q = Q_0[q:x] \cup P_3[x:c]$. By construction, Q is disjoint from P_1 and P_2 .

Now, we construct Q_1 and Q_2 given the disjoint triplet of paths (P_1, P_2, Q) such that $P_1 = (p, p_1^1 \dots, c_1)$, $P_2 = (p, p_1^2 \dots, c_2)$, and $Q = (q, q_1 \dots, c)$. We may assume that $c_1, c_2, c \in V(C)$ are pairwise distinct (cf. Figure 4.3(a)). Since G is 3-connected, there exists a path Q' connecting q and C such that $p, q_1 \notin Q'$. If Q' is disjoint from P_1, P_2 and Q we can set $Q_1 = Q$ and $Q_2 = Q'$. So, assume Q' is not disjoint from P_1, P_2, Q . Let x denote the first node along Q' (starting at q) such

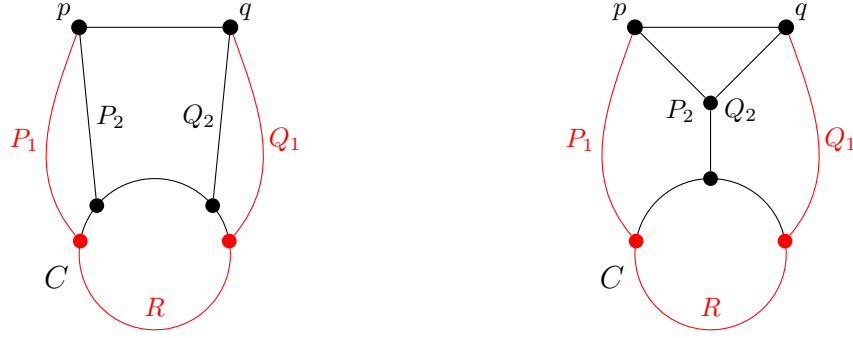
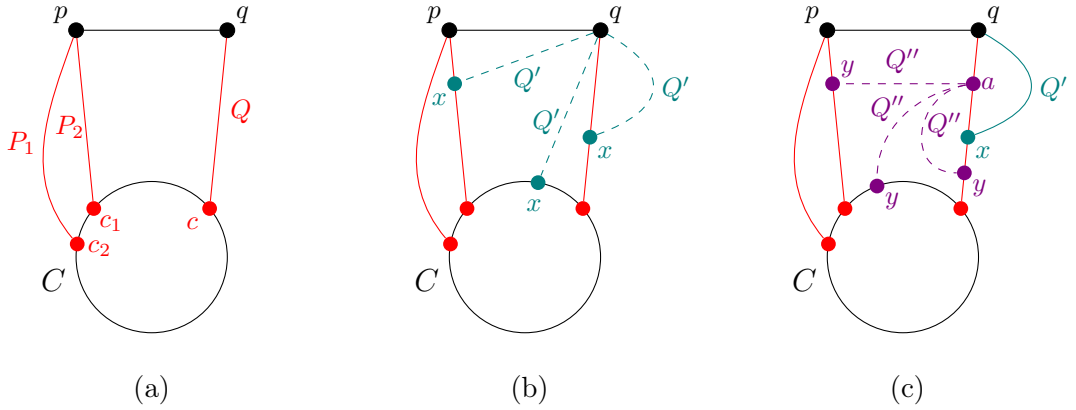
Figure 4.2: The set S (red) in the second case of the proof of Lemma 4.4.4

Figure 4.3: Visualization of obtaining the necessary paths in the second case of the proof of Lemma 4.4.4

that $x \in P_1$, $x \in P_2$, or $x \in Q$ (cf. Figure 4.3(b)). If $x \in P_i$, $i \in [2]$, we set $Q_1 = Q$ and $Q_2 = Q'[q:x] \cup P_i[x:c_2]$. Now, assume $x \in Q$. Since G is 3-connected, there exists a path Q'' connecting a node $a \in V(Q[q:x]) \setminus \{q, x\}$ with C such that the two neighbors of a in Q are not contained in Q'' . If Q'' is disjoint from P_1, P_2, Q , we set $Q_1 = Q[q:a] \cup Q''$ and $Q_2 = Q'[q:x] \cup Q[x:c]$. So, assume Q'' intersects P_1, P_2 or Q and denote by y the first such intersection along Q'' starting in a (cf. Figure 4.3(c)). We may assume $p \notin V(Q'')$ because if there was no such path, $G - \{x, p\}$ would be disconnected. Moreover, we may assume $y \notin V(Q)$ because otherwise $G - \{y, p\}$ would be disconnected. Thus (after possibly renaming), we may assume $y \in P_2[p_1^2:c_2]$. We have the claimed paths $Q_1 = Q'[q:x] \cup Q[x:c]$ and $Q_2 = Q[q:a] \cup Q''[a:y] \cup P_1[y:c_1]$. \square

Theorem 4.4.5. *Let $G = (V, E)$ be 3-connected and $C \subseteq G$ be a non-interleaved cycle. Then, $\sum_{e \in E(C)} x_e \leq 2$ defines a facet of $\text{BOND}(G)$.*

Proof. Since by Lemma 4.4.2, $\sum_{e \in E(C)} x_e \leq 2$ is valid for $\text{BOND}(G)$, there is a facet-defining inequality $a^\top x \leq b$ such that $\text{BOND}(G) \cap \{\sum_{e \in E(C)} x_e = 2\} \subseteq \text{BOND}(G) \cap \{a^\top x = b\}$. We show equality of these two faces by proving that $a_e = a_f$ for all $e, f \in E(C)$ and $a_f = 0$ for each $f \notin E(C)$.

Let $f = pq \in E(G) \setminus E(C)$. By Lemma 4.4.4 there is a set $S \subseteq V$ such that $\delta(S)$, $\delta(S \cup \{p\})$, $\delta(S \cup \{q\})$, and $\delta(S \cup \{p, q\})$ are bonds, each satisfying $\sum_{e \in E(C)} x_e = 2$ and thus, $a^\top x = b$. Hence, Lemma 4.2.9 yields $a_{pq} = 0$. Note that for each $v \in V(C)$ and $vw \in E(C)$ the bond vectors $x^{\delta(\{v\})}$ and $x^{\delta(\{v, w\})}$ satisfy $\sum_{e \in E(C)} x_e = 2$ and thus, $a^\top x = b$. Considering a 3-path on nodes v_1, \dots, v_4 labeled in order of their appearance along C , we have

$$\begin{aligned} 0 &= a^\top x^{\delta(v_2)} - a^\top x^{\delta(v_3)} = (a_{v_1 v_2} + a_{v_2 v_3}) - (a_{v_2 v_3} - a_{v_3 v_4}) = a_{v_1 v_2} - a_{v_3 v_4}, \\ 0 &= a^\top x^{\delta(\{v_2, v_3\})} - a^\top x^{\delta(v_3)} = (a_{v_1 v_2} + a_{v_3 v_4}) - (a_{v_2 v_3} - a_{v_3 v_4}) = a_{v_1 v_2} - a_{v_2 v_3}. \end{aligned}$$

Thus, we have $a_{v_1 v_2} = a_{v_2 v_3} = a_{v_3 v_4}$ yielding $a_e = a_f$ for all $e, f \in E(C)$. \square

We close this section by discussing non-interleaved cycle inequalities in 2-connected graphs. Even though validity of the non-interleaved cycle inequalities is maintained when only assuming 2-connectivity, such inequalities are not facet-defining in general for bond polytopes of non-3-connected graphs.

Example 4.4.6. Let G be the graph shown in Figure 4.4. Computing the facet description of $\text{BOND}(G)$, e.g., using the software package *Normaliz* [BIR⁺], we see the following:

- Consider the cycle induced by v_1, v_2, v_3, v_4 . The non-interleaved cycle inequality associated to this cycle is not facet-defining for $\text{BOND}(G)$.
- Consider the cycle induced by $v_1, v_2, v_3, v_7, v_6, v_5$. The non-interleaved cycle inequality associated to this cycle is facet-defining for $\text{BOND}(G)$. \blacktriangleleft

However, we can give a necessary condition for a non-interleaved cycle in a 2-connected graph giving rise to a facet-defining inequality. To do this, we call a non-interleaved cycle C *maximal non-interleaved* if there is no non-interleaved cycle C' with $|E(C) \setminus E(C')| = 1$.

Theorem 4.4.7. *Let G be a connected graph and $C \subseteq E(G)$ be a cycle. If $\sum_{e \in E(C)} x_e \leq 2$ defines a facet of $\text{BOND}(G)$, C is maximal non-interleaved.*

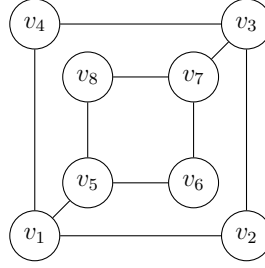


Figure 4.4: Graph from Example 4.4.6

Proof. Assume C is not maximal. Let C' be non-interleaved with $C \setminus C' = \{f\}$. Then, $\sum_{e \in E(C)} x_e \leq 2$ is the sum of the inequalities $x_f - \sum_{e \in E(C) \setminus C'} x_e \leq 0$ and $\sum_{e \in E(C')} x_e \leq 2$. By [BM86, Theorem 3.3] the first of the two inequality is valid for $\text{CUT}^\square(G)$ and thus for $\text{BOND}(G)$ and by Lemma 4.4.2 the second inequality is valid for $\text{BOND}(G)$. Hence, $\sum_{e \in E(C)} x_e \leq 2$ cannot be facet-defining for $\text{BOND}(G)$. \square

There is a class of simple non-3-connected graphs such that the non-interleaved cycle inequalities are not only facet-defining but together with the homogeneous (cut polytope) facets suffice to fully describe their bond polytopes:

Theorem 4.4.8. *For each $n \geq 3$, $\text{BOND}(C_n)$ is completely defined by the following facet-defining inequalities*

$$\begin{aligned} -x_e &\leq 0 && \text{for each } e \in E(C_n), \\ x_e - \sum_{f \in E(C_n) \setminus \{e\}} x_f &\leq 0 && \text{for each } e \in E(C_n), \\ \sum_{e \in E(C_n)} x_e &\leq 2. \end{aligned}$$

Proof. By Theorem 1.5.6, the homogeneous inequalities above are the homogeneous facets of $\text{CUT}^\square(C_n)$. Thus, by Theorem 4.1.2 these are exactly the homogeneous facets of $\text{BOND}(C_n)$. Let $\delta \subseteq E(C_n)$ be a cut. Then, δ is a bond if and only if $\delta = \emptyset$ or $|\delta| = 2$. Thus, the inequality $\sum_{e \in E(C_n)} x_e \leq 2$ defines a facet of $\text{BOND}(C_n)$.

Moreover, since the facet $\{\sum_{e \in E(C_n)} x_e = 2\} \cap \text{BOND}(C_n)$ contains all incidence vectors of non-empty bonds, each other facet has to be homogeneous. Thus, the claim follows immediately. \square

4.5 Edge- and Interleaved Cycle Inequalities

Finally, we discuss edge inequalities and a natural generalization of non-interleaved cycle inequalities. To tackle the latter, we consider the intersection of bonds and

interleaved cycles in a given graph.

Lemma 4.5.1. *Let G be a graph, $C \subseteq G$ be a cycle and $k \in \mathbb{N}$. Then, the inequality $\sum_{e \in E(C)} x_e \leq 2k$ is valid for $\text{BOND}(G)$ if and only if G does not contain a minor H of the following form: $H = T \cup T'$ where T and T' are disjoint trees, each on $k + 1$ nodes that correspond to nodes in C alternating around C . If k is chosen minimally, the inequality is tight.*

Proof. Let $\mathcal{T} \subseteq G$ be a subgraph whose contraction gives $T \subseteq H$. By adding to \mathcal{T} any components not connected to C in $G \setminus \mathcal{T}$, $\delta = \delta_G(\mathcal{T})$ becomes a bond with $|\delta \cap E(C)| = 2(k + 1)$. Conversely, if there is a bond $\delta = \delta(S) \subseteq E(G)$ with $|\delta \cap E(C)| = \ell > 2k$, ℓ even, this gives $\ell \geq 2(k + 1)$ components in $C \setminus \delta$. Since δ is a bond, both $G[S]$ and $G - S$ contain trees as minors whose nodes correspond to these components.

Now, let k be minimal such that $\sum_{e \in E(C)} x_e \leq 2k$ is valid for $\text{BOND}(G)$. Then there exists some bond δ in G such that $2(k - 1) < |\delta \cap E(C)| \leq 2k$. Tightness of the inequality follows since the number of cut edges in a cycle is always even. \square

Indeed, such inequalities are facet-defining for some graphs. One class of such graphs are generalized *Wagner graphs* V_n ($n \in 2\mathbb{N}$) also known as circulants $C_n(1, \frac{n}{2})$: V_n is obtained from the cycle C_n on nodes $[n]$ by adding the edges $\{i, i + \frac{n}{2}\}$ for $1 \leq i \leq \frac{n}{2}$. We call C_n the *outer cycle* of V_n .

Theorem 4.5.2. *Let $n \geq 6$ and C be the outer cycle of V_n . Then, the inequality $\sum_{e \in E(C)} x_e \leq 4$ defines a facet of $\text{BOND}(V_n)$.*

Proof. By Lemma 4.5.1, the inequality $\sum_{e \in E(C)} x_e \leq 4$ is valid for $\text{BOND}(V_n)$. Thus, there is a facet-defining inequality $a^\top x \leq b$ of $\text{BOND}(V_n)$ dominating it. We show $\{a^\top x = b\} = \{\sum_{e \in E(C)} x_e = 4\}$ by proving $a_f = 0$ for each $f \notin E(C)$ and $a_e = a_f$ for each $e, f \in E(C)$.

First, we show that $a_{pq} = 0$ for each $pq \notin E(C)$. For this, let v be a neighbor of p and w be a neighbor of q such that $vw \in E(V_n) \setminus E(C)$ and set $S = \{v, w\}$. Then, $\delta(S)$, $\delta(S \cup \{p\})$, $\delta(S \cup \{q\})$, and $\delta(S \cup \{p, q\})$ are bonds satisfying $\sum_{e \in E(C)} x_e = 4$ and thus, $a^\top x = b$. Hence, Lemma 4.2.9 yields $a_{pq} = 0$.

It remains to show that $a_e = a_f$ for all $e, f \in E(C)$. It suffices to prove this for two incident edges $e, f \in E(C)$. Let $\{w\} = e \cap f$, $e = vw$ and $u \in V(V_n)$ the unique node with $uw \in E(V_n) \setminus E(C)$. Then, $\delta(\{u, v\})$ and $\delta(\{u, v, w\})$ are bonds satisfying $\sum_{e \in E(C)} x_e = 4$ and thus, $a^\top x = b$. Since only edges in C have non-zero coefficients, it follows that $0 = a^\top x^{\delta(\{u, v\})} - a^\top x^{\delta(\{u, v, w\})} = a_f - a_e$. \square

On the other hand:

Example 4.5.3. Consider K_5 and a 5-cycle $C \subseteq K_5$. Then, the inequality $\sum_{e \in E(C)} x_e \leq 4$ is valid and tight but not facet-defining. ◀

Open Problem 4.5.1. Characterize interleaved cycles that induce facets.

We close this section by discussing inequalities associated to edges. By definition, the inequality $x_e \leq 1$ is always valid for $\text{BOND}(G)$. In the following, we show that although this inequality is not facet-defining in general, there is an infinite class of graphs where it is.

Lemma 4.5.4. *Let $G = (V, E)$ be a connected graph and $e \in E$. If e is contained in a non-interleaved cycle, $x_e \leq 1$ is not facet-defining for $\text{BOND}(G)$.*

Proof. Let $C \subseteq G$ be a non-interleaved cycle and $e \in E(G)$. By Lemma 4.4.2 and [BM86, Theorem 3.3], the inequalities

$$\sum_{f \in E(C)} x_f \leq 2 \quad \text{and} \quad x_e - \sum_{\substack{f \in E(C) \\ f \neq e}} x_f \leq 0$$

are valid for $\text{BOND}(G)$. Summing these two inequalities, we obtain $2x_e \leq 2$. ◻

Theorem 4.5.5. *For any $n \geq 6$ and any $e \in E(V_n)$ the inequality $x_e \leq 1$ is facet-defining for $\text{BOND}(V_n)$.*

Proof. We use the same strategy as in the proof of Theorem 4.5.2. Let $e \in E(V_n)$. Since $x_e \leq 1$ is valid for $\text{BOND}(V_n)$ there exists a facet-defining inequality $a^\top x \leq b$ dominating it. We show $\{a^\top x = b\} = \{x_e = 1\}$ by proving $a_f = 0$ for each $f \in E(V_n) \setminus \{e\}$.

First assume $f \cap e = \emptyset$. Labeling the vertices along the outer cycle by $[n]$, up to isomorphism it suffices to consider the following four cases (cf. Figure 4.5): If $e = \{1, n\}$ and $f = \{i, i+1\}$ for $2 \leq i \leq \frac{n}{2}$, we set $S^+ = [\frac{n}{2} + 1]$; if $e = \{1, n\}$ and $f = \{i, \frac{n}{2} + i\}$ for $2 \leq i \leq \frac{n}{2} - 1$, we set $S^+ = [i] \cup ([\frac{n}{2} + i] \setminus [\frac{n}{2}])$; if $e = \{\frac{n}{2}, n\}$ and $f = \{i, i+1\}$ for $1 \leq i \leq \frac{n}{2} - 2$ we set $S^+ = [i+1] \cup ([n] \setminus [\frac{n}{2} + i])$; if $e = \{\frac{n}{2}, n\}$ and $f = \{i, \frac{n}{2} + i\}$ for $1 \leq i \leq \frac{n}{2} - 1$ we set $S^+ = [i] \cup ([n] \setminus [\frac{n}{2} + i - 1])$. It is straight forward to verify that for each of these sets, S^+ , $S^+ \setminus \{i\}$, $S^+ \setminus \{j\}$, and $S^+ \setminus \{i, j\}$ (where j is the other end node of f) induce bonds satisfying $x_e = 1$ and thus $a^\top x = b$. Hence, Lemma 4.2.9 yields $a_f = 0$.

It remains to show that each edge incident to e has coefficient 0. Depending on whether e is contained in the outer cycle or not, we are in one of the situations sketched in Figure 4.6. In both cases, considering the notation as in the figure,

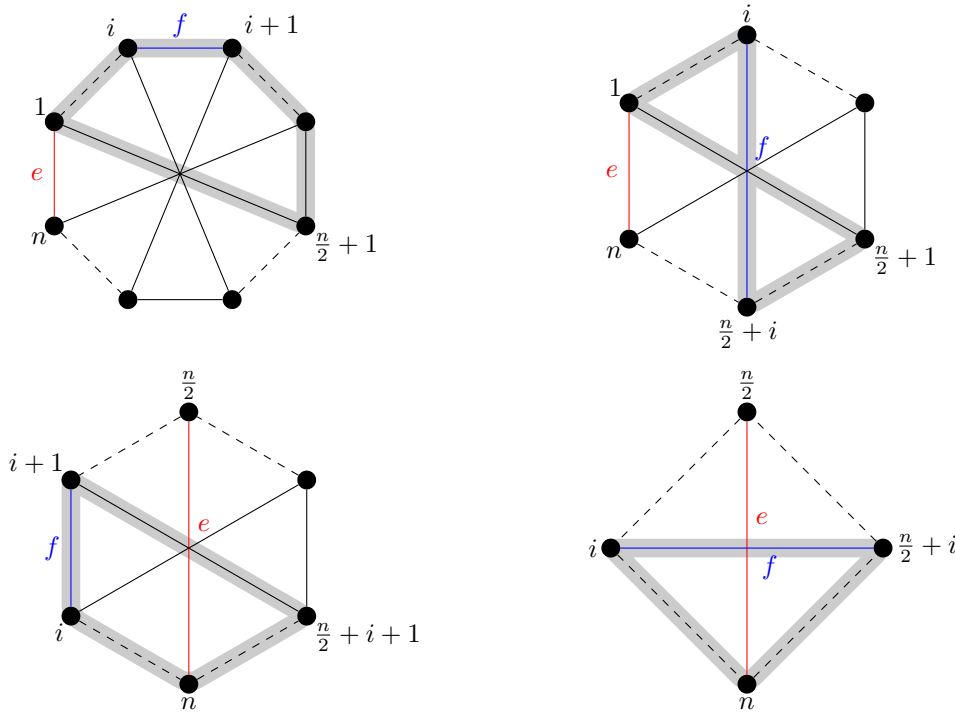


Figure 4.5: A visualization of the set S^+ from the proof of Theorem 4.5.5. The subgraph $G[S^+]$ is highlighted.

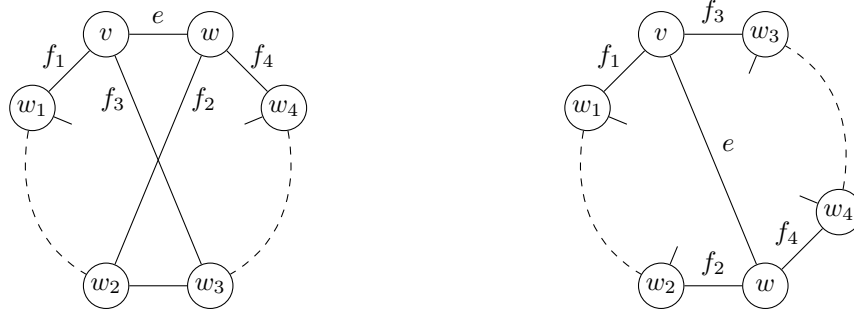
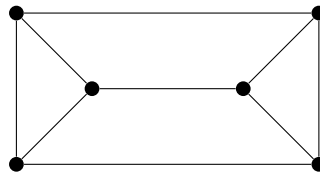
all bond-vectors in the inequalities below satisfy the equalities $x_e = 1$ and thus $a^\top x = b$. Since $a_h = 0$ for each edge $h \in E(V_n) \setminus \{e, f_1, f_2, f_3, f_4\}$, we have

$$\begin{aligned}
 b &= a^\top x^{\delta(\{v\})} &= a_e + a_{f_1} + a_{f_3}, & \quad b &= a^\top x^{\delta(\{w\})} &= a_e + a_{f_2} + a_{f_4}, \\
 b &= a^\top x^{\delta(\{v, w_1\})} &= a_e + a_{f_3}, & \quad b &= a^\top x^{\delta(\{w, w_2\})} &= a_e + a_{f_4}, \\
 b &= a^\top x^{\delta(\{v, w_3\})} &= a_e + a_{f_1}, & \quad b &= a^\top x^{\delta(\{w, w_4\})} &= a_e + a_{f_2}.
 \end{aligned}$$

Hence, we have $a_{f_1} = a^\top x^{\delta(\{v\})} - a^\top x^{\delta(\{v, w_1\})} = b - b = 0$ and analogously $a_{f_2} = a_{f_3} = a_{f_4} = 0$. \square

Given this result and noticing that each cycle in V_n is interleaved gives rise to the following question:

Open Problem 4.5.2. Let G be a 3-connected graph. Does $x_e \leq 1$ define a facet of $\text{BOND}(G)$ for each e that is not contained in a non-interleaved cycle?

Figure 4.6: Sketches of V_n with notations from the proof of Theorem 4.5.5Figure 4.7: *Prism*

4.6 $(K_5 - e)$ -Minor-Free Graphs

The focus of this section lies on $(K_5 - e)$ -minor-free graphs. We prove a linear description of bond polytopes for planar 3-connected such graphs. Moreover, we present a linear-time algorithm for MAXBOND on arbitrary $(K_5 - e)$ -minor-free graphs. We start with a characterization of these graphs.

The *wheel graph* W_n on n -nodes is obtained from the cycle C_n by adding a new node c adjacent to each node of C_n . We call c the *center node* of W_n and $C_n \subseteq W_n$ the *rim*. Moreover, we denote the graph shown in Figure 4.7 by *Prism*.

Proposition 4.6.1. [Wag60] *Each maximal $(K_5 - e)$ -minor-free graph G can be decomposed as $G = G_1 \oplus_2 \cdots \oplus_2 G_\ell$ where each G_i is isomorphic to a wheel graph, Prism, K_3 , or $K_{3,3}$.*

As a consequence, it follows that each 3-connected $(K_5 - e)$ -minor-free graph is a wheel graph, Prism, K_3 , or $K_{3,3}$. We provide a complete facet description for all planar such graphs (i.e., all but $K_{3,3}$).

Theorem 4.6.2. *Let $G \neq K_{3,3}$ be a 3-connected $(K_5 - e)$ -minor-free graph. Then*

$\text{BOND}(G)$ is completely determined by the following facet-defining inequalities:

$$\begin{aligned} x_e &\geq 0 \quad \text{for each edge } e \text{ that is not contained in a triangle,} \\ x_e - \sum_{f \in E(C) \setminus \{e\}} x_f &\leq 0 \quad \text{for each induced cycle } C \text{ and } e \in E(C), \\ \sum_{e \in E(C)} x_e &\leq 2 \quad \text{for each non-interleaved cycle } C. \end{aligned}$$

Proof. Since $\text{CUT}^\square(K_3) = \text{BOND}(K_3)$ the claim follows directly for K_3 . The description of $\text{BOND}(\text{Prism})$ can be checked by computation. Thus, it remains to prove that $\text{BOND}(W_n)$ is completely defined by the inequalities

$$\begin{aligned} x_e - x_f - x_g &\leq 0 \quad \text{for each triangle } \{e, f, g\} \text{ in } W_n, \\ x_e - \sum_{f \in R \setminus \{e\}} x_f &\leq 0 \quad \text{for each } e \in R, \\ x_e + x_f + x_g &\leq 2 \quad \text{for each triangle } \{e, f, g\} \text{ in } W_n, \\ \sum_{e \in R} x_e &\leq 2, \end{aligned}$$

where $R \subseteq E(W_n)$ denotes the set of rim edges of W_n .

By [BM86, Corollary 3.10], the homogeneous inequalities above are the homogeneous facets of $\text{CUT}^\square(W_n)$ and Theorem 4.1.2 implies that these are precisely the homogeneous facets of $\text{BOND}(W_n)$.

Let c denote the center node of W_n and $\delta \subseteq E$ be a cut. Then, δ is a bond if and only if $\delta = \emptyset$, $\delta = \delta(c)$ or $|\delta \cap R| = 2$.

Let \mathcal{P} denote the polytope given by the above inequalities. Clearly we have $\text{BOND}(W_n) \subseteq \mathcal{P}$. We prove equality of the two polytopes by showing that each vertex of \mathcal{P} is the incidence vector of some bond.

Note that by [BM86, Corollary 3.10], $\mathcal{P} = \text{CUT}^\square(W_n) \cap \{\sum_{e \in R} x_e \leq 2\}$ and the vertices of $\text{CUT}^\square(W_n)$ contained in $\{\sum_{e \in R} x_e \leq 2\}$ are exactly the incidence vectors of bonds. Now assume, \mathcal{P} has an additional vertex. Then, this is given as $\text{relint}(\mathcal{F}) \cap \{\sum_{e \in R} x_e = 2\}$ where \mathcal{F} is a 1-dimensional face of $\text{CUT}^\square(W_n)$. Such a face has to contain one of x^\emptyset and $x^{\delta(c)}$. But by Proposition 4.1.1, in $\text{CUT}^\square(W_n)$ these are only adjacent to incidence vectors of bonds which yields a contradiction. \square

Given the previous results, it seems natural to ask, whether the bond polytope of 3-connected planar graphs is completely described by inequalities associated to cycles and edges. Unfortunately the answer to this question is negative, since already $\text{BOND}(K_5 - e)$ has a facet that does not belong to the mentioned class.

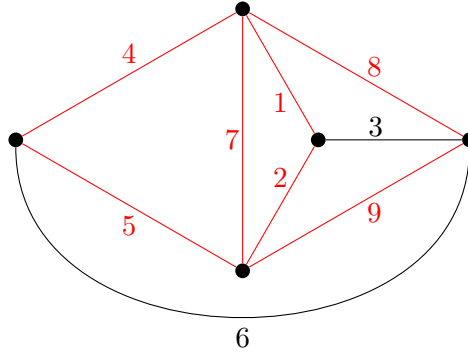


Figure 4.8: $K_5 - e$ on the edge set $E = \{1, \dots, 9\}$. Red edges are those of the support graph of the inequality from Example 4.6.3.

Example 4.6.3. Consider $K_5 - e$ with the edge labeling as in Figure 4.8. The inequality $x_1 + x_2 + x_4 + x_5 + x_7 - x_8 - x_9 \leq 2$ defines a facet of $\text{BOND}(K_5 - e)$. ◀

We close this section by presenting a linear-time algorithm for MAXBOND on $(K_5 - e)$ -minor-free graphs. For details on tree-width and parameterized algorithms see, e.g., [DF13].

Proposition 4.6.4. [DLP⁺19, DEH⁺20] Given a nice tree decomposition of G with width k , MAXBOND can be solved on G in time $2^{\mathcal{O}(k \log(k))} \times |V(G)|$.

Theorem 4.6.5. Given a $(K_5 - e)$ -minor-free graph $G = (V, E)$ with $|V| = n$, MAXBOND can be solved on G in time $\mathcal{O}(n)$.

Proof. By Proposition 4.6.1 and the fact that $Prism$, K_3 , $K_{3,3}$ are of constant size, and $|E(W_n)| = 2(|V(W_n)| - 1)$, we have $|E(G)| \in \mathcal{O}(n)$. Using Theorem 4.3.2, we can restrict ourselves to 3-connected $(K_5 - e)$ -minor-free graphs by only $\mathcal{O}(n)$ additive effort. By Proposition 4.6.1 these graphs are wheel graphs, $Prism$, K_3 , and $K_{3,3}$.

Since we can solve MAXBOND in constant time on $Prism$, K_3 , and $K_{3,3}$, it only remains to prove that MAXBOND can be solved in time $\mathcal{O}(n)$ on wheel graphs. Denoting the center of W_n by c and the rim nodes by v_1, \dots, v_n , it is straight forward to verify that a nice tree decomposition of W_n with width 3 is given by the bags $\{c\}, \{cv_1\}, \{cv_1v_2\}, \{cv_1v_2v_3\}, \{cv_1v_3\}, \{cv_1v_3v_4\}, \{cv_1v_4\}, \{cv_1v_4v_5\}, \dots, \{cv_1v_{n-1}\}, \{cv_1v_{n-1}v_n\}$. Using this tree decomposition, Proposition 4.6.4 yields the claim. ◻

Note that although the above algorithm has asymptotically linear runtime, the runtime is dependent on large constants. Since the presented tree decomposition for wheel graphs is in fact even a path decomposition and wheel graphs are of

special simple structure for this measure, it should certainly be possible to improve on the constant quite a bit. Although this might yield a more practical algorithm, this would be out of scope for this work.

4.7 Open Problems

We have introduced bond polytopes and investigated the relation of these to cut polytopes. Then, we have studied the effect of graph-operations on facets of bond polytopes. We have presented a reduction of MAXBOND to 3-connected graphs. Moreover, we have started an investigation of cycle- and edge inequalities for bond polytopes and derived a family of facet-defining inequalities for bond polytopes. Finally, we have presented a linear-time algorithm for MAXBOND on $(K_5 - e)$ -minor-free graphs as well as a linear description for all 3-connected planar such graphs.

Recall the open problems from Chapter 4.5:

Open Problem 4.5.1. Characterize interleaved cycles that induce facets.

Open Problem 4.5.2. Let G be a 3-connected graph. Does $x_e \leq 1$ define a facet of $\text{BOND}(G)$ for each e that is not contained in a non-interleaved cycle?

On the algorithmical side, we have seen the importance of clique sums. Considering cut polytopes, for $k \leq 3$ we can derive a linear description of $\text{CUT}^\square(G_1 \oplus_k G_2)$ given linear descriptions of $\text{CUT}^\square(G_1)$ and $\text{CUT}^\square(G_2)$. While we have seen how to handle 1- and 2-sums in algorithms for MAXBOND, we could not mirror this into the world of bond polytopes. As a result this would for example yield a linear description for arbitrary $(K_5 - e)$ -minor-free graphs. Therefore, the following question arises:

Open Problem 4.7.1. Given a graph $G = G_1 \oplus_k G_2$ and linear descriptions of $\text{BOND}(G_1)$ and $\text{BOND}(G_2)$. Can we derive a linear description of $\text{BOND}(G)$, at least for $k = 1, 2$?

As a first step, one may investigate how facets of $\text{BOND}(G_1)$ and $\text{BOND}(G_2)$ can be combined to obtain facets of $\text{BOND}(G)$.

Chapter 5

On the Dominant of the Multicut Polytope

This chapter is based on [CJN21].

A prominent generalization of the minimum s - t -cut problem is the *minimum multicut problem* MINMULTICUT: Given a graph G and a set $S \subseteq \binom{V(G)}{2}$ of terminal pairs, a *multicut* is an edge set $\delta \subseteq E(G)$ such that for each pair $\{s, t\} \in S$ there is no s - t -path in $G - \delta$. When the terminal set is in doubt, we may call δ an S -*multicut*. Given non-negative edge weights c_e , MINMULTICUT asks for a multicut δ minimizing $\sum_{e \in \delta} c_e$. A node $v \in V(G)$ is called a *terminal* if there exists some $w \in V(G)$ such that $\{v, w\} \in S$. A multicut is *minimal* if it is minimal with respect to inclusion, it is a *minimum multicut* if it has minimal total weight (with respect to given edge weights).

If $|S|$ is fixed, MINMULTICUT is solvable in polynomial time for $|S| = 1, 2$ [YKCP83] but NP-complete for $|S| \geq 3$ [DJP⁺94]. It remains NP-complete even when the input graph is restricted to trees of height 1, i.e., stars [GVY06]. Approximation algorithms for MINMULTICUT have been intensively studied, see e.g. [GNS06, GVY06].

We define the *multicut polytope* of G as the convex hull of all incidence vectors of multicuts, i.e.,

$$\text{MULTC}^\square(G, S) = \text{conv}(\{x^\delta : \delta \text{ is an multicut in } G \text{ with respect to } S\})$$

and the *multicut dominant* of G as

$$\text{MULTC}(G, S) = \text{MULTC}^\square(G, S) + \mathbb{R}_{\geq 0}^{E(G)}.$$

Since minimizing a non-negative objective function on $\text{MULTC}^\square(G, S)$ and $\text{MULTC}(G, S)$ yields the same result, the latter is the relevant polyhedron for the considered optimization problem.

Contrary to the cut polytope, there is only little knowledge on the multicut polytope and its dominant.

For $|S| = 1$, the multicut dominant was studied in [SW10]. Besides a characterization of vertices and adjacencies in the polyhedron, it was shown that in this case $\text{MULTC}(G, \{\{s, t\}\})$ is completely described by *edge-* and *path inequalities* (see Proposition 5.1.11 for details). Moreover, it was shown that each of these inequalities defines a facet.

Clearly this generalizes to a relaxation of $\text{MULTC}(G, S)$ for $|S| \geq 2$ by having path inequalities for each pair $\{s, t\} \in S$. In [GNS06] it was shown that when the input graph is a tree and for each $\{s, t\} \in S$ one of both nodes is a descendant of the other this relaxation coincides with $\text{MULTC}(G, S)$. Nevertheless, this does not hold in general. Already for $G = K_{1,3}$ with $S = \{\{v, w\} : v, w \text{ are leaves in } G\}$ the polyhedron defined by all edge- and path inequalities admits a fractional vertex by setting all edge variables to 0.5.

Organization of this chapter. We start in Chapter 5.1 by investigating basic properties of $\text{MULTC}(G, S)$. Moreover, we present results on liftings and projections of these polyhedra. The lifting results for the multicut dominant are stronger than those known for cut polytopes in the sense that lifting of inequalities does not only preserve validity of the inequalities but also preserves being facet-defining. This results in a characterization of facet-defining edge- and path inequalities. Then, we investigate the effect of graph operations such as node splittings and edge subdivisions on the multicut dominant and its facets in Chapter 5.2. In Chapter 5.3 we investigate facets supported on stars. In Chapter 5.4 we generalize these facet-defining inequalities to facets on trees. Both classes can be separated in polynomial time when the input graph is a tree. Finally, in Chapter 5.5 we introduce facet-defining inequalities supported on cycles.

Related Cut-Generalizations. There are multiple way to generalize cuts to different problems under the same (or similar) name in literature.

In [DGL91, DGL92] multiple polytopes associated to cut problems are studied. There, the *k-cuts* are called multicuts as well; we give give their to distinguish those from our notion of multicuts: Given a graph $G = (V, E)$ and a partition $V = S_1 \cup \dots \cup S_k$ a *k-cut* in G is the set of all edges between a node in S_i and a node in S_j for some $1 \leq i < j \leq k$.

Using this notion of multicuts, in [HLA17, LA20] the *lifted multicut problem* was studied: Given a graph G' , a subgraph $G \subseteq G'$, and a multicut $\delta \subseteq E(G)$, the lifted multicut problem asks for a minimum multicut in $\bar{\delta} \subseteq E(G')$ with $\bar{\delta} \cap E(G) = \delta$. The polytope associated to this problem is called *lifted multicut polytope*. In [LA20], the lifted multicut polytope for G being a tree or a path was studied.

5.1 Basic Properties

We start by investigating basic properties of the multicut dominant and its facet-defining inequalities. Afterwards, we study the effect of edge additions, deletions, and contractions on the multicut dominant. In particular, this leads to a classification of all facet-defining path- and edge inequalities.

The following observation is a direct consequence of the construction of the multicut dominant:

Observation 5.1.1. *Let $G = (V, E)$ be a graph and $S \subseteq \binom{V}{2}$ be a set of terminal pairs.*

- *The vertices of $\text{MULTC}(G, S)$ are precisely the incidence vectors of the (inclusion wise) minimal multicuts in G .*
- *We have $\dim \text{MULTC}(G, S) = |E|$.*
- *Let $a^\top x \geq b$ be facet-defining for $\text{MULTC}(G, S)$. Since $\text{MULTC}(G, S)$ is the Minkowski sum of a polytope and $\mathbb{R}_{\geq 0}^E$, each inner normal of a facet of $\text{MULTC}(G, S)$ is contained in $\mathbb{R}_{\geq 0}^E$, i.e. we have $a_e \geq 0$ for each $e \in E$.*
- *Let $W \supseteq V$, $\bar{G} = (W, E)$, and $\bar{S} \subseteq \binom{W}{2}$ be a set of terminal pairs such that $S = \bar{S} \cap \binom{V}{2}$. Then, we have $\text{MULTC}(G, S) = \text{MULTC}(\bar{G}, \bar{S})$.*

We can consider the support graph of facet-defining inequalities:

Lemma 5.1.2. *Let $G = (V, E)$ be a graph, $S \subseteq \binom{V}{2}$ be a set of terminal pairs, $a^\top x \geq b$ be facet-defining for $\text{MULTC}(G, S)$, and $f \in E$. Assume that $a_f \neq 0$ and $\{a^\top x \geq b\} \neq \{x_f \geq 0\}$. Then, there exists some multicut δ with $f \in \delta$ and $a^\top x^\delta = b$.*

Proof. Assume there is no such δ and let $\lambda > 0$. The inequality $a^\top x + \lambda x_f \geq b$ is valid for $\text{MULTC}(G, S)$. By assumption, each multicut δ satisfying $a^\top x = b$ satisfies $a^\top x + \lambda x_f = b$. Hence, both inequalities define the same face of $\text{MULTC}(G, S)$ contradicting the assumption that $a^\top x \geq b$ is facet-defining. \square

Theorem 5.1.3. *Let $G = (V, E)$ be a graph, $S \subseteq \binom{V}{2}$ be a set of terminal pairs, and $a^\top x \geq b$ be facet-defining for $\text{MULTC}(G, S)$ such that $\{a^\top x \geq b\} \neq \{x_{e'} \geq 0\}$ for all $e' \in E(G)$. Then, each edge $e \in E(\text{supp}(a))$ lies on an s - t -path in $\text{supp}(a)$ for some $\{s, t\} \in S$. In particular, each leaf of $\text{supp}(a)$ is a terminal.*

Proof. Assume there is some $e \in \text{supp}(a)$ such that e does not lie on any s - t -path. By Lemma 5.1.2 there is some multicut δ with $e \in \delta$ and $a^\top x^\delta = b$. Since e is not contained in any s - t -path for any $\{s, t\} \in S$, also $\delta \setminus \{e\}$ is a multicut. But, since $e \in \text{supp}(a)$, we have $a_e > 0$ and thus, $a^\top x^{\delta \setminus \{e\}} < a^\top x^\delta = b$ contradicting $a^\top x \geq b$ being valid for $\text{MULTC}(G, S)$. \square

Next, we investigate coefficients of facet-defining inequalities of the multicut dominant along *induced paths*, i.e., paths in G in which each internal node has degree 2 in G .

Theorem 5.1.4. *Let $G = (V, E)$ be a graph, $S \subseteq \binom{V}{2}$ be a set of terminal pairs, and $a^\top x \geq b$ be facet-defining for $\text{MULTC}(G, S)$ with $\{a^\top x \geq b\} \neq \{x_{e'} \geq 0\}$ for all $e' \in E$. Furthermore let $P \subseteq G$ be an induced path such that no internal node of P is a terminal. Then, $a_e = a_f$ for all $e, f \in P$.*

Proof. Assuming the contrary, let $M = \min_{e \in E(P)} a_e$, $e \in E(G)$ with $a_e = M$, and define $c \in \mathbb{R}^E$ by

$$c_e = \begin{cases} a_e, & \text{for } e \notin E(P), \\ M, & \text{for } e \in E(P). \end{cases}$$

First, we show that $c^\top x \geq b$ is valid for $\text{MULTC}(G, S)$. Let δ be a multicut in G . Since P is induced and contains no terminals as inner vertices, also $\delta' = (\delta \setminus P) \cup \{e\}$ is a multicut in G . Clearly, we have $c^\top x^\delta \geq c^\top x^{\delta'} = a^\top x^{\delta'} \geq b$.

Since $a^\top x \geq b$ is a sum of $c^\top x \geq b$ and edge inequalities $x_e \geq 0$, this contradicts the assumption that $a^\top x \geq b$ is facet-defining. \square

We want to point out that both assumptions on the path in the previous theorem are necessary. Theorem 5.4.1 will provide facet-defining inequalities having non-induced paths with different coefficients in the support graph. The facets in Theorem 5.5.3 contain induced paths with internal terminals in their support graphs and have different coefficients attached to edges in such paths.

Next, we give a complete characterization of the boundedness of facets via their support graph.

Theorem 5.1.5. *Let $G = (V, E)$ be a graph, $S \subseteq \binom{V}{2}$ be a set of terminal pairs, and $a^\top x \geq b$ be facet-defining for $\text{MULTC}(G, S)$. Then, the facet $\{a^\top x = b\} \cap \text{MULTC}(G, S)$ is bounded if and only if $\text{supp}(a) = G$.*

Proof. Since $\text{MULTC}(G, S) = \text{MULTC}^\square(G, S) + \mathbb{R}_{\geq 0}^E$, each ray in $\text{MULTC}(G, S)$ is of the form $\{y + \lambda z : \lambda \in \mathbb{R}_{\geq 0}\}$ with $y \in \text{MULTC}(G, S)$ and $z \in \mathbb{R}_{\geq 0}^E \setminus \{\mathbf{0}\}$.

If there is some edge $e \in E \setminus E(\text{supp}(a))$, we have $a^\top x = a^\top(x + x^{\{e\}})$ yielding that $\{a^\top x = b\} \cap \text{MULTC}(G, S)$ is unbounded.

Now assume that $\text{supp}(a) = G$ and there is a ray $\{y + \lambda z : \lambda \in \mathbb{R}_{\geq 0}\} \subseteq \{a^\top x = b\} \cap \text{MULTC}(G, S)$. Then, we have $0 = b - b = a^\top(y + \lambda z) - a^\top y = \lambda a^\top z$ for each $\lambda \in \mathbb{R}_{\geq 0}$. Thus, we have $a^\top z = 0$ and since $a_e > 0$ for all $e \in E$ this contradicts $z \in \mathbb{R}_{\geq 0}^E \setminus \{\mathbf{0}\}$. \square

The graph G/e is obtained from G by *contracting* the edge $e = vw$, i.e., the nodes v and w are identified, the arising self-loop is deleted and parallel edges are

merged. Considering the contraction of an edge e , there is a one-to-one correspondence between multicuts in G/e and multicuts δ in G with $e \notin \delta$. This, directly yields the following observation:

Observation 5.1.6. *Let $G = (V, E)$ be a graph, $S \subseteq \binom{V}{2}$ be a set of terminal pairs, and $e = vw \in E$ such that $\{v, w\} \notin S$. Then, $\text{MULTC}(G/e, S) = \text{MULTC}(G, S) \cap \{x_e = 0\}$.*

Next, we consider the deletion and addition of edges.

Theorem 5.1.7. *Let $G = (V, E)$ be a graph, $S \subseteq \binom{V}{2}$ be a set of terminal pairs, $H \subseteq G$, and $S' = S \cap \binom{V(H)}{2}$. Then, the following hold:*

- (i) *For $e \in E$, $\text{MULTC}(G - e, S) = \pi(\text{MULTC}(G, S))$ where $\pi: \mathbb{R}^E \rightarrow \mathbb{R}^{E \setminus \{e\}}$ is the orthogonal projection.*
- (ii) *If $\sum_{e \in E(H)} a_e x_e \geq b$ is a valid inequality for $\text{MULTC}(H, S')$, it is also valid for $\text{MULTC}(G, S)$.*
- (iii) *If $a^\top x \geq b$ is facet-defining for $\text{MULTC}(G, S)$ and $\text{supp}(a) \subseteq H$, the inequality $\sum_{e \in E(H)} a_e x_e \geq b$ is facet-defining for $\text{MULTC}(H, S')$.*
- (iv) *If $a^\top x \geq b$ is facet-defining for $\text{MULTC}(H, S')$, then $\sum_{e \in E(H)} a_e x_e \geq b$ is facet-defining for $\text{MULTC}(G, S)$.*

Proof. We start by proving (i). Then, (ii) and (iii) follow immediately. Observe that for each multicut δ in G the set $\delta \setminus \{e\}$ is a multicut in $G - e$. On the other hand, if δ' is a multicut in $G - e$, then $\delta' \cup \{e\}$ is a multicut in G . This directly yields $\text{MULTC}(G - e, S) = \pi(\text{MULTC}(G, S))$.

Now, we prove statement (iv). If $W \neq V$, we consider the graph $\bar{H} = (V, F)$. Since S' -multicuts in H and S -multicuts in \bar{H} coincide, we have $\text{MULTC}(H, S') = \text{MULTC}(\bar{H}, S)$. Thus, we may assume $W = V$. We prove the statement for the case $H^* = (V, E \setminus \{e^*\})$ for some $e^* \in E$. Then, the claim follows by adding edges in $E \setminus F$ one by one to H .

For each multicut δ in H^* , the set $\bar{\delta} = \delta \cup \{e^*\}$ is a multicut in G . Thus, $x^{\bar{\delta}}, x^{\bar{\delta}} + x^{\{e^*\}} \in \text{MULTC}(G, S)$. Since $a^\top x \geq b$ is facet-defining for $\text{MULTC}(G, S)$, lifting all multicuts satisfying $a^\top x = b$ in this fashion into the two hyperplanes $\{x_{e^*} = 1\}$ and $\{x_{e^*} = 2\}$ yields that the inequality $\sum_{e \in E \setminus \{e^*\}} a_e x_e \geq b$ defines a face of dimension at least $|E|$ of $\text{MULTC}(G, S)$. Since $a \neq \mathbf{0}$, this yields that the inequality is facet-defining. \square

Note that the facet-defining inequalities of the multicut dominant can be split into two sets: those that are also facet-defining for the multicut polytope and those that are not.

Definition 5.1.8 (shared facets). Let $G = (V, E)$ be a graph, $S \subseteq \binom{V}{2}$ be a set of terminal pairs. Given a facet-defining inequality $a^\top x \geq b$ of $\text{MULTC}(G, S)$, the defined facet is *shared* if $a^\top x \geq b$ is also facet-defining for $\text{MULTC}^\square(G, S)$.

Clearly, each facet-defining inequality $a^\top x \geq b$ of $\text{MULTC}(G, S)$ with $\text{supp}(a) = G$ is shared by Theorem 5.1.5. Moreover, considering the proof of Theorem 5.1.7, one can see that shared facets remain shared under removal of edges. Unfortunately, as we also see in that proof this does not hold for lifting in general. However, with some additional restrictions we can lift facet-defining inequalities while making sure that the property of being shared is preserved.

Lemma 5.1.9. *Let $G = (V, E)$ be a graph, $S \subseteq \binom{V}{2}$ be a set of terminal pairs, $a^\top x \geq b$ define a shared facet of $\text{MULTC}(G, S)$, $v, w \in V$ such that $e^* = vw \notin E$, and $\bar{G} = (V, E \cup \{e^*\})$. Assume there is a multicut δ^* in \bar{G} such that $e^* \notin \delta^*$ and $\sum_{e \in E} a_e x_e^\delta = b$. Then $\sum_{e \in E} a_e x_e \geq b$ defines a shared facet of $\text{MULTC}(\bar{G}, S)$.*

Proof. It follows directly from Theorem 5.1.7 (iv) that the inequality is facet-defining. Thus, it is only left to show that that the defined facet is indeed shared, i.e., that there are $|\bar{E}|$ affinely independent incidence vectors of multicuts contained in this facet.

To this end, let $m = |E|$. Since $a^\top x \geq b$ is facet-defining for $\text{MULTC}(G, S)$, there are multicuts $\delta_1, \dots, \delta_m$ in G such that $a^\top x^{\delta_i} = b$ for each $i \in [m]$ and $x^{\delta_1}, \dots, x^{\delta_m}$ are affinely independent. Now let $\bar{\delta}_i = \delta_i \cup \{e^*\}$. Then, $\bar{\delta}_i$ is a multicut in \bar{G} with $\sum_{e \in E} a_e x_e^{\bar{\delta}_i} = b$. Since $x^{\delta_1}, \dots, x^{\delta_m}$ are affinely independent and all contained in the hyperplane $\{x_{e^*} = 1\}$, the vectors $x^{\delta^*}, x^{\bar{\delta}_1}, \dots, x^{\bar{\delta}_m}$ are affinely independent and satisfy $\sum_{e \in E} a_e x_e = b$. \square

Theorem 5.1.10. *Let $G = (V, E)$ be a graph, $S \subseteq \binom{V(G)}{2}$, $H = (W, F)$ be an induced subgraph of G , and $S' = S \cap \binom{W}{2}$. Assume that for each $vw \in E \setminus F$ we have $\{v, w\} \notin S$ and there is no $\{s, t\} \in S$ with $s \in W$ and t adjacent to W . Furthermore let $a^\top x \geq b$ define a shared facet $\text{MULTC}(H, S')$. Then, $\sum_{e \in F} a_e x_e \geq b$ defines a shared facet of $\text{MULTC}(G, S)$.*

Proof. Let $G' = (V, F)$. Clearly, each S' -multicut in H is an S -multicut in G' and we thus have $\text{MULTC}(H, S') = \text{MULTC}(G', S)$.

In the following we add the edges in $E \setminus F$ one by one to G' and utilize Lemma 5.1.9 to lift the facet-defining inequality under consideration. To this end we construct for each $e \in E \setminus F$ an S -multicut $\bar{\delta}_e \subseteq E$ in G such that $e \notin \bar{\delta}_e$ and $\sum_{f \in E} a_f x_f^{\bar{\delta}_e} = b$. Since this multicut induces an according multicut in each step, this yields the claim.

Now, let $e = vw \in E \setminus F$, $\delta \subseteq F$ be an S -multicut in H (and thus in G') with $a^\top x^\delta = b$, and $\bar{\delta}_e = (\delta \cup (E \setminus F)) \setminus \{e\}$. Since H is induced and there is no $\{s, t\} \in S$

with $s \in W$ and t adjacent to W , for each $\{s, t\} \in S$ there is no s - t -path in

$$G - \bar{\delta}_e = (V \setminus (W \cup \{v, w\}), \emptyset) \cup ((H - \delta) \cup (\{v, w\}, \{e\})).$$

Thus, $\bar{\delta}_e$ is an S -multicut in G . Furthermore, we have $\sum_{f \in F} a_f x_f^{\bar{\delta}_e} = b$. \square

Finally, we investigate edge- and path inequalities for the multicut dominant. To this end, we recall the facet description for the s - t -cut dominant:

Proposition 5.1.11. *[SW10, Section 2] Let $G = (V, E)$ and $s, t \in V$. Then, the s - t -cut dominant $\text{MULTC}(G, \{\{s, t\}\})$ is completely defined by the inequalities*

$$\begin{aligned} x_e &\geq \begin{cases} 1, & \text{if } e = st, \\ 0, & \text{otherwise.} \end{cases} && \text{for all } e \in E, \\ \sum_{e \in E(P)} x_e &\geq 1, && \text{for all } s\text{-}t\text{-paths } P. \end{aligned}$$

In particular, each of these inequalities defines a facet of $\text{MULTC}(G, \{\{s, t\}\})$.

Together with Theorem 5.1.7 (iv) this yields the following:

Corollary 5.1.12. *Let $G = (V, E)$, S be a set of terminal pairs and $\{s, t\} \in S$. Then, the following hold:*

(i) *for each $vw \in E$ the inequality*

$$x_{vw} \geq \begin{cases} 1, & \text{if } \{v, w\} \in S, \\ 0, & \text{otherwise} \end{cases}$$

is facet-defining for $\text{MULTC}(G, S)$.

(ii) *For each s - t -path $P \subseteq G$ such that there does not exist $\{s', t'\} \in S$ with $s', t' \in V(P)$ and $\{s, t\} \neq \{s', t'\}$, the inequality $\sum_{e \in E(P)} x_e \geq 1$ is facet-defining for $\text{MULTC}(G, S)$*

Let $a^\top x \geq b$ be facet-defining for some $\text{MULTC}(G, S)$. In general, this inequality is not facet-defining for $\text{MULTC}(G, S')$ with $S' \supset S$.

Observation 5.1.13. *Consider a path $P = ([n], \{\{i, i+1\} : 1 \leq i < n\})$, $S' = \{(1, n)\}$, and $S = \{\{1, n\}, \{i, j\}\}$ for some $\{i, j\} \in \binom{[n]}{2} \setminus \{\{1, n\}\}$. Then $\sum_{e \in E(P)} x_e \geq 1$ is facet-defining for $\text{MULTC}(P, S')$ but not for $\text{MULTC}(P, S)$. Moreover, this carries over to arbitrary G with $P \subseteq G$.*

We close the discussion of path- and edge inequalities by investigating the relation of the polyhedron defined by these inequalities and the multicut dominant:

Lemma 5.1.14. *Let $G = (V, G)$ be a graph and $S \subseteq \binom{V}{2}$ be a set of terminal pairs. Let \mathcal{P} be the polyhedron defined by the inequalities*

$$x_{vw} \geq \begin{cases} 1, & \text{if } \{v, w\} \in S, \\ 0, & \text{else} \end{cases} \quad \text{for all } vw \in E,$$

$$\sum_{e \in E(P)} x_e \geq 1 \quad \text{for all } s\text{-}t\text{-paths } P \text{ with } \{s, t\} \in S.$$

Then, the integer points in \mathcal{P} are precisely those in $\text{MULTC}(G, S)$, i.e., $\mathcal{P} \cap \mathbb{Z}^n = \text{MULTC}(G, S) \cap \mathbb{Z}^n$.

Proof. Since each S -multicut in G is an s - t -cut in G for all $\{s, t\} \in S$, Proposition 5.1.11 yields $\text{MULTC}(G, S) \subseteq \mathcal{P}$. Now, let $x \in \mathcal{P} \cap \mathbb{Z}^E$ and define $x' \in \{0, 1\}^E$ by setting $x'_e = 1$ if $x_e \neq 0$ and $x'_e = 0$ otherwise. Since $x \in \mathcal{P}$ we have $x \in \text{MULTC}(G, \{s, t\})$ and thus, $x' \in \text{MULTC}(G, \{s, t\})$ for all $\{s, t\} \in S$. Hence, x' is incidence vector of some S -multicut in G , i.e. $x' \in \text{MULTC}(G, S)$ and thus, $x \in \text{MULTC}(G, S)$. \square

5.2 Constructing Facets from Facets

In [BM86] an extensive study on the effect of graph operations such as node splittings and edge subdivisions on the max-cut polytope and its facet-defining inequalities has been conducted. Motivated by this, we investigate the same questions for the multicut dominant. Note that in the results from [BM86] it is always mentioned that certain edges might be added to the newly obtained graph by attaching coefficient 0 to them. Since for $\text{MULTC}(G, S)$ arbitrary edges can be added by attaching weight 0 (see Theorems 5.1.7 (iv) and 5.1.10), we do not mention this explicitly in each result.

In Observation 5.1.6 we have already seen the effect of edge contractions on the multicut dominant. Now, we investigate the inverse operation:

Theorem 5.2.1 (Node splitting). *Let $G = (V, E)$ be a graph, $S \subseteq \binom{V}{2}$, $a^\top x \geq b$ be facet-defining for $\text{MULTC}(G, S)$, and $v \in \text{supp}(a)$ be a node.*

Obtain $\bar{G} = (\bar{V}, \bar{E})$ as follows: replace v by two adjacent nodes v_1 and v_2 and distribute the edges incident to v arbitrarily among v_1 and v_2 . Furthermore, obtain \bar{S} from S by replacing each pair $\{v, t\} \in S$ independently by a (not necessarily strict) subset of $\{\{v_1, t\}, \{v_2, t\}\}$.

Define $\varphi: \bar{E} \setminus \{v_1v_2\} \rightarrow E$ by

$$\varphi(e) = \begin{cases} e, & \text{if } v_1, v_2 \notin e, \\ vw, & \text{if } e = v_iw \ (i = 1, 2). \end{cases}$$

Let ω be the value of a minimum \bar{S} -multicut in $\bar{G} - v_1v_2$ when considering $a_{\varphi(e')}$, $e' \in \bar{E} \setminus \{v_1v_2\}$, as edge weights. Define $\bar{a} \in \mathbb{R}^{\bar{E}}$ by

$$\bar{a}_e = \begin{cases} b - \omega, & \text{if } e = v_1v_2, \\ a_{\varphi(e)}, & \text{otherwise.} \end{cases}$$

Then, $\bar{a}^\top x \geq b$ defines a facet of $\text{MULTC}(\bar{G}, \bar{S})$.

In particular, if $a^\top x \geq b$ defines a shared facet of $\text{MULTC}(G, S)$, so does the inequality $\bar{a}^\top x \geq b$ for $\text{MULTC}(\bar{G}, \bar{S})$.

Proof. If $a^\top x \geq b$ does not define a shared facet, utilizing Theorem 5.1.7 (iii) we remove edges $e \in E$ with coefficient $a_e = 0$ from G and add them back after splitting by applying Theorem 5.1.7 (iv). Since each facet-defining inequality is shared if its support graph is all G , we may assume that $a^\top x \geq b$ defines a shared facet. In this case we prove the ‘‘in particular’’ part.

First, we show $b - \omega \geq 0$. Note that there is a one-to-one correspondence between \bar{S} -multicuts in \bar{G} not containing v_1v_2 and S -multicut in G . Thus, since $a^\top x \geq b$ is facet-defining for $\text{MULTC}(G, S)$, there is some \bar{S} -multicut δ in \bar{G} with $\sum_{e \in \bar{E} \setminus \{v_1v_2\}} a_{\varphi(e)} x_e^\delta = b$. Since δ is also an \bar{S} -multicut in $\bar{G} - v_1v_2$, this yields $\omega \leq b$.

Next, we prove validity. Let δ be an \bar{S} -multicut in \bar{G} . If $v_1v_2 \notin \delta$, $\varphi(\delta)$ is a multicut in G ; thus, $\bar{a}^\top x^\delta = a^\top x^{\varphi(\delta)} \geq b$. Otherwise, $\delta \setminus \{v_1v_2\}$ is an S -multicut in $\bar{G} - v_1v_2$. Hence,

$$\bar{a}^\top x^\delta = (b - \omega)x_{v_1v_2} + \sum_{\substack{e \in \bar{E} \\ e \neq v_1v_2}} \bar{a}_e x_e \geq (b - \omega) + \omega = b.$$

It remains to show that $\bar{a}^\top x \geq b$ is indeed facet-defining. Since $a^\top x \geq b$ is facet-defining for $\text{MULTC}(G, S)$ there are S -multicuts $\delta_1, \dots, \delta_m$ ($m = |E|$) in G such that $x^{\delta_1}, \dots, x^{\delta_m}$ are affinely independent and $a^\top x^{\delta_i} = b$ for each $1 \leq i \leq m$. Set $\bar{\delta}_i = \varphi^{-1}(\delta_i)$. Since φ is bijective, each $\bar{\delta}_i$ is well-defined, an \bar{S} -multicut in \bar{G} , and we have $\bar{a}^\top x^{\bar{\delta}_i} = b$. Moreover, for an \bar{S} -multicut δ in $\bar{G} - v_1v_2$ with $\sum_{e \in \delta} a_e = \omega$ let $\bar{\delta} = \delta \cup \{v_1v_2\}$; we have $\bar{a}^\top x^{\bar{\delta}} = b$. Since $x^{\bar{\delta}}, x^{\bar{\delta}_1}, \dots, x^{\bar{\delta}_m}$ are affinely independent $\bar{a}^\top x \geq b$ defines a shared facet. \square

The previous theorem can be utilized to construct new classes of facet-defining inequalities from known facets, as illustrated in the following example.

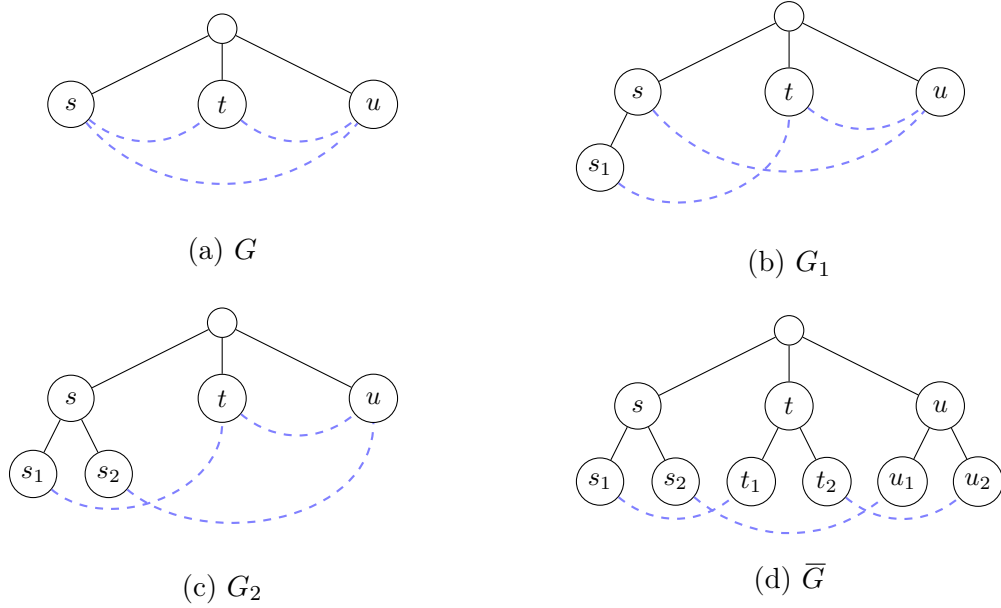


Figure 5.1: The graphs from Example 5.2.2 in black. Dashed blue connections represent the terminal pairs.

Example 5.2.2. Consider the graph G shown in Figure 5.1(a) with terminal pairs $S = \{\{s, t\}, \{s, u\}, \{t, u\}\}$. We will later see in Theorem 5.3.3 that $\sum_{e \in E(G)} x_e \geq 2$ is facet-defining for $\text{MULTC}(G, S)$.

Consider the graph G_1 obtained from G by splitting s into s and s_1 (cf. Figure 5.1(b)) and a new set of terminal pairs $S_1 = \{\{s_1, t\}, \{s, u\}, \{t, u\}\}$. Then Theorem 5.2.1 yields that $\sum_{e \in E(G_1)} x_e \geq 2$ defines a facet of $\text{MULTC}(G_1, S_1)$. Now, we obtain G_2 from G_1 by a splitting of s into s and s_2 (cf. Figure 5.1(c)) and setting $S_2 = \{\{s_1, t\}, \{s_2, u\}, \{t, u\}\}$; Theorem 5.2.1 yields that $\sum_{e \in E(G_2)} x_e \geq 2$ is facet-defining for $\text{MULTC}(G_2, S_2)$. Splitting t and u in the same fashion, we obtain the graph \bar{G} shown in Figure 5.1(d) with $\bar{S} = \{\{s_1, t_1\}, \{s_2, u_1\}, \{t_2, u_2\}\}$. By Theorem 5.2.1, $\sum_{e \in E(\bar{G})} x_e \geq 2$ is facet-defining for $\text{MULTC}(\bar{G}, \bar{S})$. ◀

Theorem 5.2.3 (Edge subdivision). *Let $G = (V, E)$ be a graph, $S \subseteq \binom{V}{2}$, $a^\top x \geq b$ be facet-defining for $\text{MULTC}(G, S)$, and $f \in E$. Obtain $\bar{G} = (\bar{V}, \bar{E})$ from G by subdividing f into f_1, f_2 . Then,*

$$a_f(x_{f_1} + x_{f_2}) + \sum_{e \in E \setminus \{f\}} a_e x_e \geq b \quad (5.1)$$

is facet-defining for $\text{MULTC}(\bar{G}, S)$.

In particular, if $a^\top x \geq b$ defines a shared facet of $\text{MULTC}(G, S)$, so does (5.1) for $\text{MULTC}(\overline{G}, S)$.

Proof. Validity of the inequality is straight-forward to verify. Let $m = |E|$. As in the proof of Theorem 5.2.1 we may assume that $a^\top x \geq b$ defines a shared facet and prove the ‘‘in particular’’ statement for this case.

Since $a^\top x \geq b$ defines a shared facet of $\text{MULTC}(G, S)$, there exist multicuts $\delta_1, \dots, \delta_m$ in G such that $x^{\delta_1}, \dots, x^{\delta_m}$ are affinely independent and $a^\top x^{\delta_i} = b$ for all $i \in [m]$. Now set for each $i \in [m]$

$$\overline{\delta}_i = \begin{cases} \delta_i, & \text{if } f \notin \delta_i, \\ (\delta_i \setminus \{f\}) \cup \{f_1\}, & \text{otherwise.} \end{cases}$$

By Lemma 5.1.2 we may without loss of generality assume $f \in \delta_1$ and set $\delta = (\delta_1 \setminus \{f\}) \cup \{f_2\}$. Then, $\delta, \overline{\delta}_1, \dots, \overline{\delta}_m$ are multicuts in \overline{G} . Since $x^{\delta_1}, \dots, x^{\delta_m}$ are affinely independent and $x_{f_2}^\delta = 1$ whereas $x_{f_1}^{\overline{\delta}_1} = \dots = x_{f_2}^{\overline{\delta}_m} = 0$, the vectors $x^\delta, x^{\overline{\delta}_1}, \dots, x^{\overline{\delta}_m}$ are affinely independent. Moreover, they all satisfy (5.1) with equality. Hence, inequality (5.1) defines a shared facet of $\text{MULTC}(\overline{G}, S)$. \square

Iteratively applying Theorem 5.2.3 we obtain the following corollary:

Corollary 5.2.4 (Replacing an edge by a path). *Let $G = (V, E)$ be a graph, $S \subseteq \binom{V}{2}$, $a^\top x \geq b$ be facet-defining for $\text{MULTC}(G, S)$, and $f \in E$. Obtain $\overline{G} = (\overline{V}, \overline{E})$ by replacing f by a path P . Then,*

$$a_f \sum_{e \in E(P)} x_e + \sum_{e \in \overline{E} \setminus E(P)} a_e x_e \geq b \quad (5.2)$$

is facet-defining for $\text{MULTC}(\overline{G}, S)$.

In particular, if $a^\top x \geq b$ defines a shared facet of $\text{MULTC}(G, S)$, so does (5.2) for $\text{MULTC}(\overline{G}, S)$.

If we replace an edge $uv \in E(G)$ in a graph G by any connected graph H , Theorem 5.1.7 (iv) yields a facet for each facet-defining inequality of $\text{MULTC}(G, S)$ and any uv -path in H . Next, we want to study the inverse operation, i.e., the replacement of certain subgraphs by an edge.

Theorem 5.2.5 (Replacing a connected graph by an edge). *Let $G = (V, E)$ be a graph and $S \subseteq \binom{V}{2}$. Let $H \subseteq G$ be a connected subgraph of G such that H shares precisely two vertices s, t with the rest of G , i.e., no edge in $E(G) \setminus E(H)$ is incident to any node in $V(H) \setminus \{s, t\}$. Assume there is no terminal in $V(H) \setminus \{s, t\}$. Let $a^\top x \geq b$ be facet-defining for $\text{MULTC}(G, S)$ and let ω be the weight of a minimum*

s - t -cut δ_{st} in H with respect to edge weights given by a . Obtain $\bar{G} = (\bar{V}, \bar{E})$ from G by replacing H by the edge st and define $\bar{a} \in \mathbb{R}^{\bar{E}}$ by

$$\bar{a}_e = \begin{cases} a_e, & \text{if } e \neq st, \\ \omega, & \text{if } e = st. \end{cases}$$

Then, $\bar{a}^\top x \geq b$ is facet-defining for $\text{MULTC}(\bar{G}, S)$.

Proof. By Theorem 5.1.7 (iii) and (iv), we may remove all edges in G with coefficient 0, and add such edges in $E(G) \setminus E(H)$ back after the replacement of H . Hence, we may assume that $G = \text{supp}(a)$ and $\omega \neq 0$. We start by proving validity of the claimed inequality. Let δ be a multicut in \bar{G} . If $st \notin \delta$, then δ is also a multicut in G and thus $\bar{a}^\top x^\delta = a^\top x^\delta \geq b$. If $st \in \delta$, the set $(\delta \setminus \{st\}) \cup \delta_{st}$ is an S -multicut in G and we have $\bar{a}^\top x^\delta = a^\top x^{(\delta \setminus \{st\})} \geq b$.

Now, let $\pi: \mathbb{R}^E \rightarrow \mathbb{R}^{\bar{E}}$ be the projection given by

$$\pi(x)_e = \begin{cases} x_e, & \text{if } e \neq st, \\ \frac{1}{\omega} \sum_{e \in E(H)} x_e, & \text{if } e = st. \end{cases}$$

Clearly, we have $\bar{a}^\top \pi(x^\delta) = b$ for each S -multicut δ in G with $a^\top x^\delta = b$. Thus,

$$\begin{aligned} & \dim(\{\bar{a}^\top x = b\} \cap \text{MULTC}(\bar{G}, S)) \\ & \geq \dim(\{a^\top x = b\} \cap \text{MULTC}(G, S)) - (|E(H)| - 1) = |\bar{E}|. \end{aligned}$$

Moreover, since $a^\top x \geq b$ is facet-defining we have $a \neq \mathbf{0}$ and thus, $\bar{a} \neq \mathbf{0}$. Hence, $\bar{a}^\top x \geq b$ is facet-defining for $\text{MULTC}(\bar{G}, S)$. \square

Corollary 5.2.4 and Theorem 5.2.5 naturally give rise to the following conjecture:

Conjecture 5.2.6. *Let $G = (V, E)$ be a graph, $S \subseteq \binom{V}{2}$, and \bar{G} be obtained from G by replacing an edge by a connected graph. Then, any facet-defining inequality of $\text{MULTC}(\bar{G}, S)$ is either an edge inequality, or obtained from facet-defining inequalities of $\text{MULTC}(G, S)$ by applying Corollary 5.2.4.*

5.3 Star Inequalities

In this section we investigate the multicut dominant of $K_{1,n}$ with respect to specific sets of terminal pairs. For $K_{1,3}$ with leaves W and $S = \binom{W}{2}$, it is known that edge- and path inequalities do not suffice to give a complete description of $\text{MULTC}(K_{1,3}, S)$ [GVY06, Section 1]. Since S can be viewed as a triangle on the leaves, there are two natural generalizations of this instance to $K_{1,n}$: The terminal

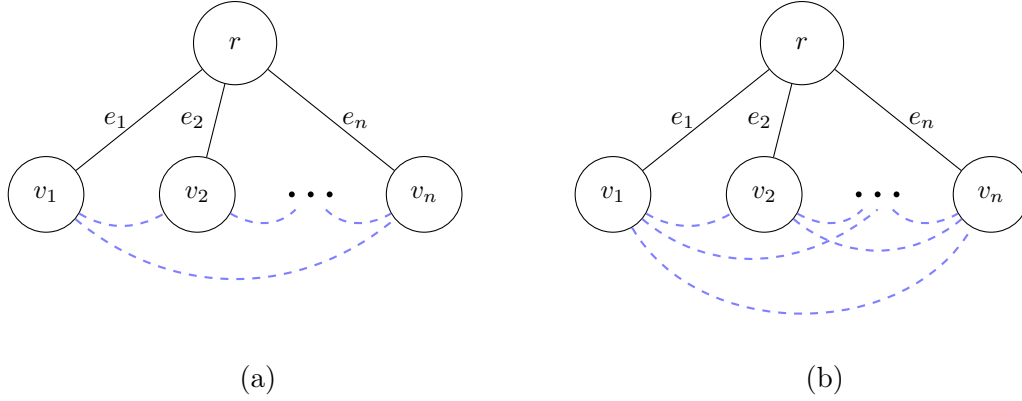


Figure 5.2: The graphs from Theorems 5.3.1 and 5.3.3 in black. Dashed blue connections represent the terminal pairs.

pairs may form a cycle on the leaves or the terminal pairs may form a complete graph on the leaves. We discuss facets based on both generalizations.

Through this section we consider a star $K_{1,n} = (\{r, v_0, \dots, v_{n-1}\}, \{0, \dots, n-1\})$ with $i = rv_i$ for $0 \leq i < n$.

Theorem 5.3.1. *Let $n \geq 3$ be odd, consider $K_{1,n}$, and let $S = \{\{v_i, v_{(i+1) \bmod n}\} : 0 \leq i < n\}$ (cf. Figure 5.2(a)). Then, the circular n -star inequality $\sum_{i=0}^{n-1} x_i \geq \lceil \frac{n}{2} \rceil$ defines a shared facet of $\text{MULTC}(K_{1,n}, S)$.*

Proof. It is straight-forward to verify validity of the inequality. We now prove that it is indeed facet-defining. For $0 \leq k < n$ set $\delta_k^n = \{(k + 2i) \bmod n : 0 \leq i \leq \lfloor \frac{n}{2} \rfloor\}$. Then, the δ_k^n are pairwise distinct multicut in $K_{1,n}$ as each such set contains precisely two consecutive (but different) edges of $K_{1,n}$. Clearly, it holds $\sum_{j=1}^{n-1} x_j^{\delta_k^n} = \lceil \frac{n}{2} \rceil$. We show by induction that $x^{\delta_0^n}, \dots, x^{\delta_{n-1}^n}$ are affinely independent. Clearly, this is true for $n = 3$.

For $2 \leq i < n$ we have $\pi(x^{\delta_k^n}) = x^{\delta_{k-2}^{n-2}}$ where $\pi: \mathbb{R}^n \rightarrow \mathbb{R}^{n-2}$ is the orthogonal projection $(x_0, \dots, x_{n-1}) \mapsto (x_2, \dots, x_{n-1})$. Thus, by induction we know that $x^{\delta_2^n}, \dots, x^{\delta_{n-1}^n}$ are affinely independent. Let $\lambda_0, \dots, \lambda_{n-1} \in \mathbb{R}_{\geq 0}$ with $\sum_{k=0}^{n-1} \lambda_k = 0$ and $0 = \sum_{k=0}^{n-1} \lambda_k x^{\delta_k^n}$. Comparing the coefficients of the first three entries of this

affine combination, we obtain

$$\begin{aligned}\lambda_0 + \sum_{k=1}^{\lfloor n/2 \rfloor} \lambda_{2k-1} &= 0 \\ \lambda_1 + \sum_{k=1}^{\lfloor n/2 \rfloor} \lambda_{2k} &= 0 \\ \lambda_0 + \lambda_2 + \sum_{k=2}^{\lfloor n/2 \rfloor} \lambda_{2k-1} &= 0\end{aligned}$$

Subtracting the first from the third inequality we obtain $\lambda_2 - \lambda_1 = 0$ and summing the first and second inequality we obtain $0 = \lambda_1 + \sum_{i=0}^{n-1} \lambda_i = \lambda_1$. Thus, $\lambda_1 = \lambda_2 = 0$. Since $x^{\delta_3^n}, \dots, x^{\delta_n^n}$ are affinely independent, we conclude $\lambda_1 = \dots = \lambda_n = 0$.

The defined facet is shared since the inequality is minimally supported on $K_{1,n}$. \square

Note that considering even n in the scenario of the previous theorem, the corresponding inequality $\sum_{e \in E(K_{1,n})} x_e \geq \frac{n}{2}$ would be dominated by path inequalities and thus not be facet-defining. However, the inequalities from the previous theorem together with edge- and path inequalities suffice to completely describe the multicut dominant for arbitrary n .

Corollary 5.3.2. *Let $n \geq 3$, consider $K_{1,n}$, and let $S = \{\{v_i, v_{(i+1) \bmod n}\} : 0 \leq i < n\}$. Then, the $\text{MULTC}(K_{1,n}, S)$ is completely described by the inequalities*

$$\begin{aligned}x_e &\geq 0 && \text{for all } e \in E, \\ x_i + x_{(i+1) \bmod n} &\geq 1 && \text{for } 0 \leq i < n, \\ \sum_{e \in E} x_e &\geq \left\lceil \frac{n}{2} \right\rceil,\end{aligned}$$

where the last inequality can be omitted if and only if n is odd.

Proof. A matrix A is *totally unimodular* if each square submatrix has determinant -1 , 0 , or 1 . It is well-known that if A is totally unimodular, and $b \in \mathbb{Z}^m$, all vertices of the polyhedron $\{x \in \mathbb{R}^d : Ax \geq b\}$ are integral.

Let $\mathcal{P} = \{Ax \geq b\}$ be the polyhedron defined by the claimed inequalities. In the following we prove that each vertex of \mathcal{P} is integral yielding $\mathcal{P} = \text{MULTC}(G, S)$.

First assume n is even. We show that A is totally unimodular by using a characterization due to [HT16, Appendix]: Let A be an $(m \times n)$ -matrix such that every entry of A equals -1 , 0 , or 1 . Then A is totally unimodular if the columns of A can be partitioned into two disjoint sets B and C such that

- every row of A contains at most two non-zero entries;
- if two non-zero entries in the same row have the same sign, the column of one is in B and the other in C ; and
- if two non-zero entries in the same row have opposite signs, either both columns are in B or both columns are in C .

By partitioning the columns of A based on the parity of their index, the previous characterization yields that A is totally unimodular. Thus, all vertices of \mathcal{P} are integral.

Now, assume n is odd. Let $v \in \mathcal{P}$ be a vertex. We call the above path inequalities *2-path inequalities* due to the length of their support. Note that the only point in \mathbb{R}^E satisfying all 2-path inequalities with equality is $(\frac{1}{2}, \dots, \frac{1}{2})$, which is not contained in \mathcal{P} . Thus, there is a 2-path inequality that is not satisfied by v with equality. By symmetry we may assume that this 2-path inequality is $x_n + x_1 \geq 1$. Let A' and b' be obtained from A and b , respectively, by deleting the row corresponding to this inequality. Then, v is a vertex of the polyhedron $\{A'x \geq b'\}$. We prove $v \in \mathbb{Z}^{E(K_{1,n})}$ by showing that A' is totally unimodular. To this end, we use a characterization from [FG65, Section 8]: Let A be an $(m \times n)$ -matrix such that every entry of A equals 0 or 1. Then, A is totally unimodular if its columns can be permuted such that for every row the 1s appear consecutively.

A' is (up to permutation of rows) of the form

$$\begin{pmatrix} 1 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 & 1 \\ 1 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & 1 \\ 1 & 1 & \cdots & 1 & 1 \end{pmatrix}$$

and thus, totally unimodular. \square

We now turn our attention to the second mentioned class of instances. cf. Figure 5.2(b).

Theorem 5.3.3. *Let $n \in \mathbb{N}$, consider $K_{1,n}$ and let $S = \{\{v_i, v_j\} : 0 \leq i < j < n\}$. Then, the complete n -star inequality $\sum_{e \in E(K_{1,n})} x_e \geq n - 1$ defines a shared facet of $\text{MULTC}(K_{1,n}, S)$.*

Proof. It is straight-forward to verify that the inequality is valid for $\text{MULTC}(G, S)$. Moreover, the vertices of $\text{MULTC}(K_{1,n}, S)$ are $x^{E \setminus \{i\}}$ for all $0 \leq i < n$. These are affinely independent. Since all of these vectors satisfy $\sum_{i=0}^{n-1} x_i = n - 1$, the inequality is facet-defining.

The defined facet is shared since the inequality is minimally supported on $K_{1,n}$. \square

Together with Theorem 5.1.7 (iv) this theorem gives a facet-defining inequality for $\text{MULTC}(K_{1,n}, S)$ with $S = \{\{s, t\} : s, t \text{ are leaves}\}$ corresponding to each $K_{1,k} \subseteq K_{1,n}$ with the induced sets of terminals. Thus, the previous theorem gives a large number of facet-defining inequalities for $\text{MULTC}(K_{1,n}, S)$. Motivated by this we generalize these inequalities further in Chapter 5.4 by considering more general trees instead of stars.

By [GVY06] the minimum multicut problem is NP-hard on trees (in fact already on stars). Motivated by this, we present a polynomial-time separation algorithm for generalizations of circular- and complete n -star inequalities for the multicut dominants when the input graph is restricted to a tree. We call the inequalities obtained from these facet-defining inequalities by applying Corollary 5.2.4 *subdivided circular n -star inequalities* and *subdivided complete n -star inequalities*, respectively. By Theorem 5.1.7 (iv) they yield facet-defining inequalities for each graph containing the according subdivision of $K_{1,n}$ with respective sets of terminal pairs.

Corollary 5.3.4. *Let $k \in \mathbb{N}$ be fixed. Given an input graph G that is a tree and a set $S \subseteq \binom{V(G)}{2}$ of terminals, we can enumerate all facet-defining subdivided circular k -stars and subdivided complete k -stars inequalities for $\text{MULTC}(G, S)$ in polynomial time.*

In particular, these inequalities can thus be separated in polynomial time.

Proof. We prove that all facet-defining subdivided complete k -star inequalities can be enumerated in polynomial time. This can then be shown analogously for subdivided circular k -star inequalities. Checking each enumerated inequality individually yields a simple separation routine.

Let $G = (V, E)$ be a tree with $|V| = n$. There are $n \cdot \binom{n-1}{k} \in \mathcal{O}(n^{k+1})$ choices for a root $r \in V$ and nodes $v_1, \dots, v_k \in V \setminus \{r\}$. We check in linear time (in $|V| + |S|$) whether these nodes form the root and leaves of a $K_{1,k}$ subdivision in G by searching for the unique r - v_i -paths in G while checking whether these paths are disjoint and no terminal pair containing a node different from v_1, \dots, v_k is induced. Then, we can verify whether the leaves induce the necessary terminal pairs in S . Hence, we obtain an overall runtime of $\mathcal{O}((|V| + |S|)^{k+2})$ \square

Although enumerating all such inequalities might not be very practical, this result should be considered as a proof of concept. We are convinced that there are

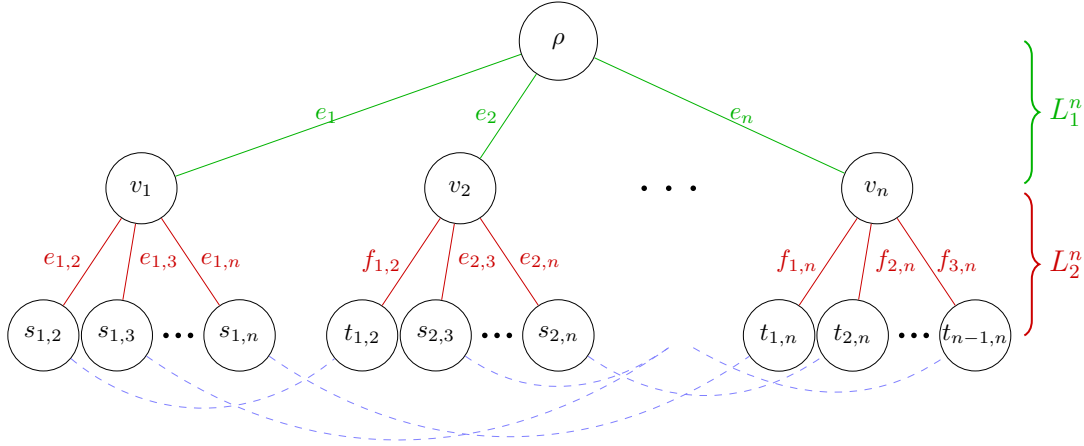


Figure 5.3: The tree T_n . Edges in L_1^n are green, edges in L_2^n are red. Dashed blue connections visualize S_n

more efficient separation routines for these inequalities using more sophisticated algorithmic approaches. However, this discussion would be out of scope for this work.

5.4 Tree Inequalities

As we saw in Example 5.2.2, the star inequalities can be generalized to facet-defining inequalities on trees by applying node splits. In the following, we further investigate these inequalities.

Throughout this section we consider the graph T_n , as showcased in Figure 5.3: T_n is a rooted tree on $n^2 + 1$ nodes: The root r has n children v_1, \dots, v_n and there are leaves $s_{i,j}, t_{i,j}$ ($1 \leq i < j \leq n$) such that $s_{i,j}$ is a child of v_i and $t_{i,j}$ is a child of v_j . For $i, j \in [n]$, we set $e_i = rv_i$, $e_{i,j} = v_i s_{i,j}$, and $f_{i,j} = v_j t_{i,j}$. Moreover, we let $L_1^n = \{e_1, \dots, e_n\}$ and $L_2^n = \{e_{i,j}, f_{i,j} : 1 \leq i < j \leq n\}$. Finally, let $S_n = \{\{s_{i,j}, t_{i,j}\} : 1 \leq i < j \leq n\}$. Observe that $|L_1^n| = n$ and $|L_2^n| = 2\binom{n}{2}$; G thus has precisely n^2 edges. The main goal of this section is to prove the following theorem:

Theorem 5.4.1. *For all $n > k \geq 2$, the (n, k) -tree inequalities*

$$(n - k) \sum_{e \in L_1^n} x_e + \sum_{e \in L_2^n} x_e \geq k(n - k) + \binom{n - k}{2} \tag{5.3}$$

define shared facets of $\text{MULTC}(T_n, S_n)$.

To prove this theorem we first prove two auxiliary lemmata.

Lemma 5.4.2. *For $n > k \geq 2$, inequality (5.3) is valid for $\text{MULTC}(T_n, S_n)$. In particular, the solutions for which (5.3) is tight are precisely the minimal multicuts δ in G with $|\delta \cap L_1^n| \in \{k-1, k\}$.*

Proof. Let δ be a minimal multicut with $|\delta \cap L_1^n| = \ell$. Since there are $\binom{n-\ell}{2}$ terminal pairs not separated by $\delta \cap L_1^n$ and the removal of an edge in L_2^n separates at most one of those pairs we have $|\delta \cap L_2^n| = \binom{n-\ell}{2}$. Thus,

$$\begin{aligned} & (n-k) \sum_{e \in L_1^n} x_e^\delta + \sum_{e \in L_2^n} x_e^\delta - \left(k(n-k) + \binom{n-k}{2} \right) \\ &= \ell(n-k) + \binom{n-\ell}{2} - k(n-k) - \binom{n-k}{2} \\ &= \frac{1}{2}(k-\ell-1)(k-\ell) \geq 0. \end{aligned}$$

Where the last inequality holds since $k, \ell \in \mathbb{N}$. The *in particular* part follows since the above inequality is satisfied with equality if and only if $\ell \in \{k-1, k\}$. \square

The following lemma considers the case $k = 2$ of (5.3).

Lemma 5.4.3. *For $n \geq 3$ and $k = 2$, inequality (5.3) is facet-defining for $\text{MULTC}(T_n, S_n)$.*

Proof. For the reader's ease, we rewrite the inequality under consideration as

$$(n-2) \sum_{e \in L_1^n} x_e + \sum_{e \in L_2^n} x_e \geq 2(n-2) + \binom{n-2}{2} \quad (5.4)$$

Validity of (5.4) is shown in Lemma 5.4.2. It remains to verify that the inequality is indeed facet-defining. To this end, we show by induction over n that there are n^2 S_n -multicuts with affinely independent incidence-vectors satisfying (5.4) with equality. For $n = 3$, this follows from Example 5.2.2.

Now, let $n \geq 4$. By induction there exist S_{n-1} -multicuts $\delta'_1, \dots, \delta'_{(n-1)^2}$ in T_{n-1} such that $\{x^{\delta'_i} : 1 \leq i \leq (n-1)^2\}$ is an affine independent set and all $x^{\delta'_i}$ satisfy the equality $((n-1)-2) \sum_{e \in L_1^{n-1}} x_e^{\delta'_i} + \sum_{e \in L_2^{n-1}} x_e^{\delta'_i} = 2((n-1)-2) + \binom{(n-1)-2}{2}$.

Let $\delta_i = \delta'_i \cup \{e_{j,n} : j \in [n], e_j \notin \delta'_i\}$ for $1 \leq i \leq (n-1)^2$. Since $x^{\delta'_1}, \dots, x^{\delta'_{(n-1)^2}}$ are affinely independent and $|E(T_{n-1})| = (n-1)^2$, for each $\ell \in [n-1]$ there is some $i_\ell \in [(n-1)^2]$ with $e_\ell \notin \delta'_{i_\ell}$. Setting $\widehat{\delta}_\ell = \delta'_{i_\ell} \cup \{e_{j,n} : j \in [n] \setminus \{\ell\}, e_j \notin \delta'_{i_\ell}\} \cup \{f_{\ell,n}\}$ for $1 \leq \ell < n$ the set

$$A = \left\{ x^{\delta_i} : 1 \leq i \leq (n-1)^2 \right\} \cup \left\{ x^{\widehat{\delta}_\ell} : 1 \leq \ell < n \right\}$$

is affinely independent and each $x \in A$, attains equality in (5.4). Now, for $i \in [n-1]$ let $\gamma_i = \{e_i, e_n\} \cup \{e_{a,b} : a, b \neq i, 1 \leq a < b < n\}$ and let $\gamma_n = \{e_n\} \cup \{e_{a,b} : 1 \leq a < b < n\}$. Then, x^{γ_i} attains equality in (5.4). We prove that $A \cup \{x^{\gamma_1}, \dots, x^{\gamma_n}\}$ is affinely independent. Since $|A \cup \{x^{\gamma_1}, \dots, x^{\gamma_n}\}| = n^2$ this yields the claim. Note that $A \subseteq \{x_n = 0\}$ and $x^{\gamma_1} \notin \{x_n = 0\}$. Thus, $A \cup \{x^{\gamma_1}\}$ is affinely independent. Now, assume that $A \cup \{x^{\gamma_1}, \dots, x^{\gamma_k}\}$ is affinely independent. Let $\mathcal{H} = \left\{ \sum_{i=1}^k x_i + (k-1)x_n + \sum_{i=1}^k (x_{e_{i,n}} + x_{f_{i,n}}) = k \right\}$. By construction, we have

$$A \cup \{x^{\gamma_1}, \dots, x^{\gamma_k}\} \subseteq \mathcal{H} \quad \text{and} \quad x^{\gamma_{k+1}} \notin \mathcal{H}.$$

Thus, $A \cup \{x^{\gamma_1}, \dots, x^{\gamma_{k+1}}\}$ is affinely independent and the claim follows by induction. \square

Given the previous two lemmata, we can now prove the main theorem of this section. As a tool we use the following simple observation from linear algebra:

Observation 5.4.4. *Let $n > k \geq 1$ and let $\mathbf{1}_i \in \mathbb{R}^n$ be the i -th unit-vector. Then, there exist $M_1, \dots, M_n \in \binom{[n]}{k}$ such that $\sum_{i \in M_1} \mathbf{1}_i, \dots, \sum_{i \in M_n} \mathbf{1}_i$ are linearly independent.*

Proof of Theorem 5.4.1. Validity of (5.3) is proven in Lemma 5.4.2. We know from Lemma 5.4.3 that the claim holds for any pair $(n, 2)$ with $n \geq 3$. Using this as the basis for our induction, it suffices to show that the claim for the pair $(n+1, k+1)$ follows from the truth of the statement for (n, k) . Thus, for the induction step assume that

$$(n-k) \sum_{e \in L_1^n} x_e + \sum_{e \in L_2^n} x_e \geq k(n-k) \binom{n-k}{2} \quad (5.5)$$

is facet-defining for $\text{MULTC}(T_n, S_n)$. We show that

$$((n+1) - (k+1)) \sum_{e \in L_1^{n+1}} x_e + \sum_{e \in L_2^{n+1}} x_e \geq (k+1)((n+1) - (k+1)) \binom{(n+1) - (k+1)}{2} \quad (5.6)$$

is facet-defining for $\text{MULTC}(T_{n+1}, S_{n+1})$.

Let $n+1 \geq 4$ and $k+1 \geq 3$. To prove the induction step, we construct $(n+1)^2$ S_{n+1} -multicuts with affinely independent incidence vectors each choosing k or $k+1$ edges in L_1^{n+1} . By Lemma 5.4.2 these incidence vectors satisfy (5.6).

Since by induction hypothesis (5.5) is facet-defining for $\text{MULTC}(T_n, S_n)$, there exist S_n -multicuts $\delta'_1, \dots, \delta'_{n^2}$ in T_n such that $x^{\delta'_1}, \dots, x^{\delta'_{n^2}}$ are affinely independent and attain equality. Thus, setting $\delta_i = \delta'_i \cup \{e_{n+1}\}$ for $1 \leq i \leq n$, the set $X^\delta = \{x^{\delta_1}, \dots, x^{\delta_n}\}$ is affinely independent and satisfy (5.6) with equality.

By Observation 5.4.4 there exist $A_1, \dots, A_n \in \binom{[n]}{n-k-1}$ such that the vectors $\sum_{i \in A_1} \mathbb{1}_i, \dots, \sum_{i \in A_n} \mathbb{1}_i$ are linearly independent. Thus, setting

$$\gamma_i = \{e_j : j \in [n] \setminus A_i\} \cup \{e_{i,j} : 1 \leq j < k \leq n, j, k \in A_i\} \cup \{e_{j,n+1} : j \in A_i\} \quad \text{and} \\ \gamma'_i = \{e_j : j \in [n] \setminus A_j\} \cup \{e_{i,j} : 1 \leq j < k \leq n, j, k \in A_i\} \cup \{f_{j,n+1} : j \in A_i\}$$

for $1 \leq i \leq n$ the set $X^\gamma = \{x^{\gamma_1}, \dots, x^{\gamma_n}, x^{\gamma'_1}, \dots, x^{\gamma'_n}\}$ is linearly independent. Since $x_{e_{j,n+1}}^{\delta_i} = x_{f_{j,n+1}}^{\delta_i} = 0$ for each $1 \leq i \leq n^2$ and $1 \leq j \leq n$, the set $X^\delta \cup X^\gamma$ is affinely independent.

Finally, we set $\delta = \{e_i : 1 \leq i \leq k\} \cup \{e_{i,j} : k+1 \leq i < j \leq n+1\}$ and $\mathcal{H} = \{(n-k-1)x_{e_{n+1}} + \sum_{i \in [n]} x_{e_{i,n+1}} + x_{f_{i,n+1}} = n-k-1\}$. Since $X^\delta \cup X^\gamma \subseteq \mathcal{H}$ and $x^\delta \notin \mathcal{H}$, the set $X^\delta \cup X^\gamma \cup \{x^\delta\}$ is affinely independent.

Since the inequality is supported on T_n , the facet is shared. \square

Motivated by the NP-hardness of MINMULTICUT when the input graph is restricted to a tree, we present a polynomial-time separation algorithm for generalizations of (n, k) -tree inequalities for the multicut dominant in this case. We call the inequalities obtained from these facet-defining inequalities by applying Corollary 5.2.4 *subdivided (n, k) -tree inequalities*. By Theorem 5.1.7 (iv) these yield facet-defining inequalities for each graph containing the according subdivision of T_n with respective sets of terminal pairs. As before, we consider our separation algorithm as a proof of concept and are convinced that there are more efficient separation routines utilizing more sophisticated algorithmic approaches whose discussion would be out of scope for this work.

Corollary 5.4.5. *Let $\ell \in \mathbb{N}$ be fixed. Given an input graph G that is a tree and a set $S \subseteq \binom{V(G)}{2}$ of terminal pairs, we can enumerate all facet-defining subdivided (ℓ, k) -tree inequalities for $\text{MULTC}(G, S)$ in polynomial time.*

In particular, these inequalities can thus be separated in polynomial time.

Proof. We prove that facet-defining subdivided (ℓ, k) -tree inequalities can be enumerated in polynomial time. Checking each enumerated inequality individually yields a simple separation routine.

Let $G = (V, E)$ be a tree with $|V| = n$. There are $n \cdot \binom{n-1}{\ell} \cdot \binom{n-1-\ell}{2 \cdot \binom{\ell}{2}} \in \mathcal{O}(n^{2\ell^2+\ell+1})$ choices for a root $r \in V$, nodes $v_1, \dots, v_\ell \in V \setminus \{r\}$, and nodes $s_{i,j}, t_{i,j} \in V \setminus \{r, v_1, \dots, v_\ell\}$, $1 \leq i < j \leq \ell$. We check in linear time (in $|V| + |S|$) whether these nodes form the respective nodes of a T_n subdivision in G by searching for the unique r - v_i -, v_i - $s_{i,j}$ -, and v_i - $t_{i,j}$ -paths ($1 \leq i < j \leq \ell$) in G while checking whether all these paths are disjoint and no terminal pair containing nodes which are not in $\{s_{i,j}, t_{i,j} : 1 \leq i < j \leq \ell\}$ is induced. If they do, we can verify whether the leaves induce the necessary terminal pairs in S . Since ℓ is fixed, the number of inequalities corresponding to this tree is constant. Hence, we obtain an overall runtime of $\mathcal{O}((|V| + |S|)^{2\ell^2+\ell+2})$. \square

5.5 Cycle Inequalities

After the investigation of facets of the multicut dominant supported on stars and trees, naturally the question arises whether there are also facet-defining inequalities with 2-connected support. We provide a positive answer to this question by introducing two classes of facet-defining inequalities supported on cycles. Throughout this section we consider the cycle $C_n = (\{v_0, \dots, v_{n-1}\}, \{0, \dots, n-1\})$ with $i = v_i v_{(i+1) \bmod n}$ for $0 \leq i < n$.

First, we consider cycles with each non-edge being a terminal pair:

Theorem 5.5.1. *Let $n \geq 5$ be odd, consider C_n and the set of terminal pairs $S = \{\{v, w\} : vw \notin E(C_n)\}$. Then, the inequality*

$$\sum_{e \in E(C_n)} x_e \geq \left\lceil \frac{n}{2} \right\rceil$$

defines a shared facet of $\text{MULTC}(C_n, S)$.

Proof. To prove validity of the inequality let δ be a multicut in C_n . Then, δ intersects each 2-path of C_n . Since there are n 2-paths and each edge is contained in two such paths, we obtain $2 \cdot |\delta| \geq n$ and thus, $|\delta| \geq \left\lceil \frac{n}{2} \right\rceil$.

To show that the inequality is indeed facet-defining, consider the multicuts $\delta_i = \{(i + 2k) \bmod n : 1 \leq k < \frac{n}{2}\}$ for $1 \leq i \leq n$. Clearly, we have $\sum_{e \in \delta_i} x_e = \left\lceil \frac{n}{2} \right\rceil$. The affine independence of these incidence vectors was shown in the proof of Theorem 5.3.1.

Since the inequality is minimally supported on C_n , the facet is shared. \square

Corollary 5.5.2. *Let $n \geq 5$, consider C_n and the set of terminal pairs $S = \{\{v, w\} : vw \notin E(C_n)\}$. Then, $\text{MULTC}(C_n, S)$ is completely described by the inequalities*

$$\begin{aligned} x_e &\geq 0 && \text{for all } e \in E(C_n), \\ x_{uv} + x_{vw} &\geq 1 && \text{for all } uv, vw \in E(C_n), \\ \sum_{e \in E(C_n)} x_e &\geq \left\lceil \frac{n}{2} \right\rceil, \end{aligned}$$

where the last inequality can be omitted if and only if n is odd.

Proof. We can reuse the proof of Corollary 5.3.2 since the polyhedra coincide despite arising from different instances. \square

Finally, we consider another, less dense set of terminal pairs over a cycle. There, the graph and terminals form a Moebius ladder instead of a complete graph, cf. Figure 5.4.

Theorem 5.5.3. *Let $n \geq 5$ be odd, consider C_{2n} and the set of terminal pairs $S = \{\{v_i, v_{i+n}\} : 1 \leq i \leq n\}$. Then, for $\beta \in \{1, 2\}$ and $\beta' = 3 - \beta$, the inequalities*

$$\sum_{i=0}^{n-1} (\beta x_{2i-1} + \beta' x_{2i}) \geq 3$$

define shared facets of $\text{MULTC}(C_{2n}, S)$.

Proof. We prove that the inequality $\sum_{i=1}^n (x_{2i-1} + 2x_{2i}) \geq 3$ defines a shared facet of $\text{MULTC}(C_{2n}, S)$. Then, the claim follows for $\beta = 2$ by symmetry.

Validity follows from the fact that the only feasible multicuts with less than three edges are $\delta = \{i, i+n\}$ for $0 \leq i < n$.

It remains to prove that the inequality is indeed facet-defining. To this end consider the multicuts

$$\begin{aligned} \delta_i &= \{i, i+n\} && \text{for } 0 \leq i < n, \\ \gamma_i &= \left\{ (2(i+\ell)) \bmod n : \ell \in \left\{ 0, 1, \left\lfloor \frac{n}{2} \right\rfloor \right\} \right\} && \text{for } 0 \leq i < n. \end{aligned}$$

Clearly, the incidence vectors of each such multicuts satisfies the inequality under consideration with equality. Hence, it remains to prove that these multicuts are affinely independent. This is trivial for $x^{\delta_1}, \dots, x^{\delta_n}$. Moreover, since $x_j^{\gamma_i} = 0$ for each i and each odd j it suffices to prove that $x^{\gamma_1}, \dots, x^{\gamma_n}$ are affinely independent. To this end we consider the matrix $A = (a_{i,j})_{0 \leq i, j \leq n}$ with entries $a_{i,j} = x_{2i}^{\gamma_j}$ and show that A has full rank.

Assume A has rank less than n . Since A is a circulant matrix, [Fer57, Theorem 9] together with $\det A = \det A^T$ yields

$$0 = \det A = \prod_{j=0}^{n-1} \left(1 + \zeta^{(n-2)j} + \zeta^{\frac{n-1}{2}j} \right)$$

where $\zeta = \exp(\frac{2i\pi}{n})$. Then, we have $1 + \zeta^{(n-2)j} + \zeta^{\frac{n-1}{2}j} = 0$ for some $0 \leq j \leq n$. Now, we have $\{\zeta^{(n-2)j}, \zeta^{\frac{n-1}{2}j}\} = \{\exp(\frac{2i\pi}{3}), \exp(\frac{4i\pi}{3})\}$. Hence, we have $\zeta^{(n-2)j} = (\zeta^{\frac{n-1}{2}j})^2 = \zeta^{(n-1)j}$ and thus, $j \bmod n = 0$ contradicting $1 + \zeta^{(n-2)j} + \zeta^{\frac{n-1}{2}j} = 0$.

Since the inequality minimally is supported on C_{2n} , the defined facet is shared. \square

Theorem 5.5.3 does not have a natural variant for even n . However, we can generalize its inequalities by node splittings in the following way:

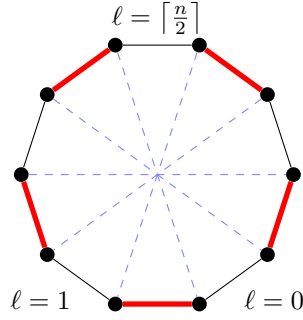


Figure 5.4: The graph from Theorem 5.5.3 for $n = 5$. Dashed blue connections visualize the set S , black edges are those with coefficient 1, and thick, red edges are those with coefficient 2. We label the edges of a potential γ_i -cut

Theorem 5.5.4. *Let $n \geq 5$ be odd, $N \geq n$, and $0 < \ell_1 < \dots < \ell_{n-1} < \ell_n = N$. Consider the cycle C_{2N} and $S_{2N} = \{\{v_i, v_{i+N}\} : 1 \leq i \leq N\}$. Then, for $\beta \in \{1, 2\}$ and $\beta' = 3 - \beta$,*

$$\sum_{i=0}^{\ell_1-1} (\beta x_i + \beta' x_{i+N}) + \sum_{i=\ell_1}^{\ell_2-1} (\beta' x_i + \beta x_{i+N}) + \dots + \sum_{i=\ell_{n-1}}^{\ell_n-1} (\beta x_i + \beta' x_{i+N}) \geq 3$$

define shared facets of $\text{MULTC}(C_{2N}, S_{2N})$.

Proof. We prove by induction on $N - n$ that the inequality is indeed facet-defining for $\beta = 1$. Since the inequality is minimally supported on C_{2n} , the defined facet is then shared. The claim follows symmetrically for $\beta = 2$.

For $N = n$, the claim is given by Theorem 5.5.3. Now, let $N > n$. By induction and symmetry we may assume that

$$\sum_{i=0}^{\ell_1-1} (x_i + 2x_{i+N}) + \sum_{i=\ell_1}^{\ell_2-1} (2x_i + x_{i+N}) + \dots + \sum_{i=\ell_{n-1}}^{\ell_n-1} (x_i + 2x_{i+N}) \geq 3$$

is facet-defining for $\text{MULTC}(C_{2(N-1)}, S_{2(N-1)})$. We now construct the claimed facet-defining inequality by using two node splits utilizing Theorem 5.2.1, cf. Figure 5.5. First we obtain the graph G from $C_{2(N-1)}$ by splitting v_{N-1} into v_{N-1} and v'_{N-1} such that v_{N-1} is adjacent to v_N and v'_{N-1} is adjacent to v_{N-2} , and set $S = S_{2(N-1)} \setminus \{\{v_0, v_{N-1}\}\} \cup \{\{v_0 v_{N-1}\}, \{v_0, v'_{N-1}\}\}$. A minimum S -multicut in $G - v_{N-1} v'_{N-1}$ with respect to the coefficient from the inequality under consideration has value 2 and is witnessed by $\{0, \ell_1 + N\}$ as the former edge separates all terminal pairs

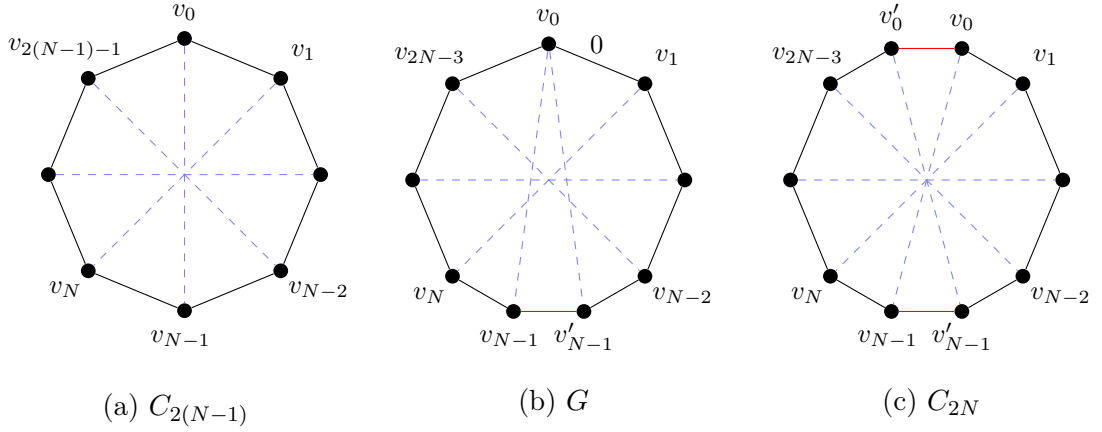


Figure 5.5: Visualization of the node splitting in the proof of Theorem 5.5.4. Dashed blue connections indicate terminal pairs. Edges obtained by node splitting are colored red.

but $\{v_0, v_{N-1}\}$, which is separated by the latter edge. Hence, Theorem 5.2.1 yields that

$$1 \cdot x_{v_{N-1}v'_{N-1}} + \sum_{i=0}^{\ell_1-1} (x_i + 2x_{i+N}) + \sum_{i=\ell_1}^{\ell_2-1} (2x_i + x_{i+N}) + \cdots + \sum_{i=\ell_{n-1}}^{\ell_n-1} (x_i + 2x_{i+N}) \geq 3 \quad (5.7)$$

is facet-defining for $\text{MULTC}(G, S)$.

Now, we obtain C_{2N} from G by splitting v_0 into v_0 and v'_0 such that v_0 is adjacent to v_1 and v'_0 is adjacent to v_{2N-3} . Within this split we obtain $S_{2N} = S_{2(N-1)} \setminus \{\{v_0, v_{N-1}\}\} \cup \{\{v_0v_{N-1}\}, \{v'_0, v'_{N-1}\}\}$. Since a minimum S_{2N} -multicut in $C_{2N} - v_0v'_0$ with respect to the coefficient from (5.7) is given by $\{v_{N-1}v'_{N-1}\}$ and has value 1, Theorem 5.2.1 yields that

$$2x_{v_0v'_0} + x_{v_{N-1}v'_{N-1}} + \sum_{i=0}^{\ell_1-1} (x_i + 2x_{i+N}) + \sum_{i=\ell_1}^{\ell_2-1} (2x_i + x_{i+N}) + \cdots + \sum_{i=\ell_{n-1}}^{\ell_n-1} (x_i + 2x_{i+N}) \geq 3$$

is facet-defining for $\text{MULTC}(C_{2N}, S_{2N})$. After renaming, this is the claimed facet-defining inequality. \square

Given the inequalities from Theorems 5.5.3 and 5.5.4, the question arises naturally whether these, together with edge- and path inequalities, suffice to describe the multicut dominant of C_{2n} . We were able to verify this computationally using normaliz [BIR⁺] for $n \leq 7$ and end with the following conjecture:

Conjecture 5.5.5. *Let $n \geq 3$, consider C_{2n} , and $S = \{\{v_i, v_{i+n}\} : 1 \leq i \leq n\}$. Then, $\text{MULTC}(C_{2n}, S)$ is completely defined by the edge inequalities, the path inequalities, and the inequalities from Theorems 5.5.3 and 5.5.4.*

5.6 Open Problems

We introduced the multicut dominant and investigated its basic properties. Moreover, we studied the effect of graph operations on the multicut dominant and its facets. We presented facet-defining inequalities supported on stars, trees, and cycles.

Apart from Conjectures 5.2.6 and 5.5.5, there are open research questions regarding constraint separation algorithms that we have not tackled in this chapter: Are there more efficient separation routines for subdivided star- and tree-equalities on trees than those given in Corollaries 5.3.4 and 5.4.5? Can one give a polynomial separation method for some class of the presented facet-defining inequalities of the multicut dominant of arbitrary graphs?

It is known that `MINMULTICUT` is solvable in polynomial time when restricted to $|S| \leq 2$. For $|S| = 1$, this is mirrored on the polyhedral side by the complete description of the s - t -cut dominant. All our new facet-defining inequalities require instances with at least 3 terminal pairs. This gives rise to the following conjecture:

Conjecture 5.6.1. *Let $G = (V, E)$ and S be a set of terminal pairs with $|S| = 2$. Then, `MULTC`(G, S) is completely described by the following inequalities:*

$$\begin{aligned} x_e &\geq 0, && \text{for all } e \in E, \\ \sum_{e \in (P)} x_e &\geq 1, && \text{for each } s\text{-}t\text{-path } P \text{ with } \{s, t\} \in S. \end{aligned}$$

Chapter 6

Conclusion

In this thesis we investigated different cut problems—namely the maximum cut problem, the maximum bond problem, and the minimum multicut problem—as well as their associated polyhedra. The chapters roughly fall into two categories: While the introduction of a fixed parameter tractable algorithm for maximum cut parameterized by the crossing number in Chapter 3 was purely algorithmic, Chapter 2, 4, and 5 mainly focused on polyhedra. In the following, we highlight main results from the latter three sections and state some related open problems. For further open problems arising from each individual chapter, we point the reader to Chapter 2.3, 3.4, 4.7, and 5.6.

In Chapter 2, we completely characterized graphs admitting a simple or simplicial cut polytope. In this light it would be interesting to start an investigation on the polyhedral structure of bond polytopes and multicut dominants as objects of discrete geometry.

In Chapters 2 and 4, we introduced linear-time reductions to 3-connectivity for the maximum cut problem and the maximum bond problem. Moreover, we gave complete linear descriptions for cut polytopes of $K_{3,3}$ -minor-free graphs and bond polytopes of 3-connected planar $(K_5 - e)$ -minor-free graphs. In contrast to the NP-completeness of these problems on general graphs, we showed that the maximum cut problem can be solved in polynomial time on $K_{3,3}$ -minor-free graphs and that the maximum bond problem can be solved in linear time on $(K_5 - e)$ -minor-free graphs. For the multicut problem, however, we cannot hope to find polynomial-time algorithms or complete linear descriptions for some (reasonable) classes of graphs obtained by excluding a specific minor, as this problem is already NP-complete when the input graphs are restricted to stars.

While the investigation of basic properties of bond polytopes in Chapter 4 yielded mainly negative answers, the respective study of the multicut dominant in Chapter 5 gave rise to nice lifting- and projection results. Regarding the effect of graph operations on the bond polytope, the multicut dominant, and their respec-

tive facet-defining inequalities, we were able to transfer many known results from the study of cut polytopes and also work out important discrepancies.

Moreover, we presented new classes of facet-defining inequalities for the bond polytope and the multicut dominant. While finding separation algorithms for the inequalities for the bond polytope and cycle inequalities for the multicut dominant remains an open problem, we showed that star- and tree inequalities for the multicut dominant can in fact be separated in polynomial time when the input graph is a tree. Utilizing our star- and tree inequalities and (maybe more efficient) separation algorithms could yield powerful solvers for the minimum multicut problem.

Bibliography

- [AI07] D. Avis and T. Ito. New classes of facets of the cut polytope and tightness of I_{mm22} Bell inequalities. *Discrete Applied Mathematics*, 155(13):1689–1699, 2007.
- [Bar82] F. Barahona. On the computational complexity of Ising spin glass models. *Journal of Physics A: Mathematical and General*, 15(10):3241–3253, 1982.
- [Bar83] F. Barahona. The max-cut problem on graphs not contractible to K_5 . *Operations Research Letters*, 2(3):107–111, 1983.
- [BB97] R. Battiti and A.A. Bertossi. Differential greedy for the 0-1 equicut problem. In *Network Design: Connectivity and Facilities Location, Proceedings of a DIMACS Workshop, Princeton, New Jersey, USA, April 28-30, 1997*, volume 40 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 3–21. DIMACS/AMS, 1997.
- [BCN⁺21] D. Bokal, M. Chimani, A. Nover, J. Schierbaum, T. Stolzmann, M.H. Wagner, and T. Wiedera. Properties of large 2-crossing-critical graphs. *arXiv*, abs/2112.04854, 2021.
- [BCR97] L. Brunetta, M. Conforti, and G. Rinaldi. A branch-and-cut algorithm for the equicut problem. *Mathematical Programming*, 77:243–263, 1997.
- [BFL⁺16] H.L. Bodlaender, F.V. Fomin, D. Lokshtanov, E. Penninkx, S. Saurabh, and D.M. Thilikos. (Meta) Kernelization. *Journal of the ACM*, 63(5):44:1–44:69, 2016.
- [BG09] W. Bruns and J. Gubeladze. *Polytopes, rings, and K-theory*. Springer Monographs in Mathematics. Springer, Dordrecht, 2009.
- [BGJR88] F. Barahona, M. Grötschel, M. Jünger, and G. Reinelt. An application of combinatorial optimization to statistical physics and circuit layout design. *Operations Research*, 36(3):493–513, 1988.
- [BGM85] F. Barahona, M. Grötschel, and A.R. Mahjoub. Facets of the bipartite subgraph polytope. *Mathematics of Operations Research*, 10(2):340–358, 1985.
- [BIR⁺] W. Bruns, B. Ichim, T. Römer, R. Sieg, and C. Söger. Normaliz. algorithms for rational cones and affine monoids. <https://normaliz.uos.de>.
- [BM86] F. Barahona and A.R. Mahjoub. On the cut polytope. *Mathematical Programming*, 36(2):157–173, 1986.
- [Bot93] R.A. Botafogo. Cluster analysis for hypertext systems. In *Proceedings of the 16th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Pittsburgh, PA, USA, June 27 - July 1, 1993*, pages 116–125. ACM, 1993.

- [CCG⁺13] R. Carvajal, M. Constantino, M. Goycoolea, J. Vielma, and A. Weintraub. Imposing connectivity constraints in forest planning models. *Operations Research*, 61:824–836, 2013.
- [CDJ⁺19] M. Chimani, C. Dahn, M. Juhnke-Kubitzke, N.M. Kriege, P. Mutzel, and A. Nover. Maximum cut parameterized by crossing number. *arXiv*, abs/1903.06061, 2019.
- [CDJ⁺20] M. Chimani, C. Dahn, M. Juhnke-Kubitzke, N.M. Kriege, P. Mutzel, and A. Nover. Maximum cut parameterized by crossing number. *Journal Graph Algorithms and Applications*, 24(3):155–170, 2020.
- [CH17] M. Chimani and P. Hliněný. A tighter insertion-based approximation of the crossing number. *Journal of Combinatorial Optimization*, 33:1183–1225, 2017.
- [Cha17] B. Chaourar. A linear time algorithm for a variant of the MAX CUT problem in series parallel graphs. *Advances in Operations Research*, 2017:1267108:1–1267108:4, 2017.
- [Cha20] B. Chaourar. Connected max cut is polynomial for graphs without the excluded minor $K_5 \setminus e$. *Journal of Combinatorial Optimization*, 40(4):869–875, 2020.
- [CJN20] M. Chimani, M. Juhnke-Kubitzke, and A. Nover. On the bond polytope. *arXiv*, abs/2012.06288, 2020.
- [CJN21] M. Chimani, M. Juhnke-Kubitzke, and A. Nover. On the dominant of the multicut polytope. *arXiv*, abs/2112.01095, 2021.
- [CJNR22] M. Chimani, M. Juhnke-Kubitzke, A. Nover, and T. Römer. Cut polytopes of minor-free graphs. *Bulletin mathématique de la Société des Sciences Mathématiques de Roumanie*, (tba), 2022. (see <https://arxiv.org/abs/1903.01817>).
- [CKMV19] M. Chimani, P. Kindermann, F. Montecchiani, and P. Valtr. Crossing numbers of beyond-planar graphs. In *Graph Drawing and Network Visualization - 27th International Symposium, GD 2019, Prague, Czech Republic, September 17-20, 2019, Proceedings*, pages 78–86, 2019.
- [CKP⁺07] J. Chen, I.A. Kanj, L. Perkovic, E. Sedgwick, and G. Xia. Genus characterizes the complexity of certain graph problems: Some tight results. *Journal of Computer and System Sciences*, 73(6):892–907, 2007.
- [CLS11] D.A. Cox, J.B. Little, and H.K. Schenck. *Toric Varieties*. Graduate studies in mathematics. American Mathematical Society, 2011.
- [CR01] T. Christof and G. Reinelt. Decomposition and parallelization techniques for enumerating the facets of combinatorial polytopes. *International Journal of Computational Geometry & Applications*, 11(04):423–437, 2001.
- [dBt96] G. di Battista and R. Tamassia. On-line planarity testing. *SIAM Journal on Computing*, 25:956–997, 1996.
- [DDW16] G. Ding, S. Dziobiak, and H. Wu. Large W_k - or $K_{3,t}$ -minors in 3-connected graphs. *Journal of Graph Theory*, 82(2):207–217, 2016.
- [DEH⁺20] G. L. Duarte, H. Eto, T. Hanaka, Y. Kobayashi, Y. Kobayashi, D. Lokshantov, L. L. C. Pedrosa, R. C. S. Schouery, and U. S. Souza. Computing the largest bond and the maximum connected cut of a graph. *arXiv*, abs/2007.04513, 2020.

- [DF13] R. G. Downey and M. R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer London, 2013.
- [DGL91] M. Deza, M. Grötschel, and M. Laurent. Complete descriptions of small multicut polytopes. *Applied Geometry and Discrete Mathematics*, 4, 01 1991.
- [DGL92] M. Deza, M. Grötschel, and M. Laurent. Clique-web facets for multicut polytopes. *Mathematics of Operations Research - MOR*, 17:981–1000, 11 1992.
- [Die90] R. Diestel. *Graph Decompositions: A Study in Infinite Graph Theory*. Oxford science publications. Clarendon Press, 1990.
- [Die18] R. Diestel. *Graph theory*, volume 173 of *Graduate Texts in Mathematics*. Springer, Berlin, fifth edition, 2018.
- [DJP⁺94] E. Dahlhaus, D. Johnson, C. Papadimitriou, P. Seymour, and M. Yannakakis. The complexity of multiterminal cuts. *SIAM Journal on Computing*, 23:864–894, 08 1994.
- [DKM18] C. Dahn, N.M. Kriege, and P. Mutzel. A fixed-parameter algorithm for the Max-Cut problem on embedded 1-planar graphs. In *Combinatorial Algorithms - 29th International Workshop, IWOCA 2018, Singapore, July 16-19, 2018, Proceedings*, pages 141–152, 2018.
- [DL92a] M. Deza and M. Laurent. Facets for the cut cone I. *Mathematical Programming*, 56:121–160, 1992.
- [DL92b] M. Deza and M. Laurent. Facets for the cut cone II: clique-web inequalities. *Mathematical Programming*, 56:161–188, 1992.
- [DL94a] M. Deza and M. Laurent. Applications of cut polyhedra I. *Journal of Computational and Applied Mathematics*, 55(2):191–216, 1994.
- [DL94b] M. Deza and M. Laurent. Applications of cut polyhedra II. *Journal of Computational and Applied Mathematics*, 55(2):217–247, 1994.
- [DL10] M. Deza and M. Laurent. *Geometry of cuts and metrics*, volume 15 of *Algorithms and Combinatorics*. Springer, Heidelberg, 2010.
- [DLP⁺19] G. L. Duarte, D. Lokshtanov, L. L. C. Pedrosa, R. C. S. Schouery, and U. S. Souza. Computing the largest bond of a graph. In *Proc. of Intl Symposium on Parameterized and Exact Computation, IPEC 2019*, volume 148 of *LIPICs*, pages 12:1–12:15, 2019.
- [DS16] M. Deza and M.D. Sikirić. Enumeration of the facets of cut polytopes over some highly symmetric graphs. *International Transactions in Operational Research*, 23(5):853–860, 2016.
- [DSDJ⁺95] C. De Simone, M. Diehl, M. Jünger, P. Mutzel, G. Reinelt, and G. Rinaldi. Exact ground states of Ising spin glasses: New experimental results with a branch-and-cut algorithm. *Journal of Statistical Physics*, 80(1-2):487–496, 1995.
- [dSL95] C.C. de Souza and M. Laurent. Some new classes of facets for the equicut polytope. *Discrete Applied Mathematics*, 62(1-3):167–191, 1995.
- [DT98] M. Dell’Amico and M. Trubian. Solution of large weighted equicut problems. *European Journal of Operational Research*, 106(2-3):500–521, 1998.

- [EFF04] J.A. Ellis, H. Fan, and M.R. Fellows. The dominating set problem is fixed parameter tractable for graphs of bounded genus. *Journal of Algorithms*, 52(2):152–168, 2004.
- [EHKK19] H. Eto, T. Hanaka, Y. Kobayashi, and Y. Kobayashi. Parameterized algorithms for maximum cut with connectivity constraints. In *Proc. of Intl Symposium on Parameterized and Exact Computation, IPEC 2019*, volume 148 of *LIPICs*, pages 13:1–13:15, 2019.
- [Eng11] A. Engström. Cut ideals of K_4 -minor free graphs are generated by quadrics. *Michigan Mathematical Journal*, 60(3):705–714, 2011.
- [Fer57] W. L. Ferrar. *Algebra: A Text-book of Determinants, Matrices, and Algebraic Forms*. Oxford University Press, 1957.
- [FF56] L. R. Ford and D. R. Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 8:399–404, 1956.
- [FG65] D. R. Fulkerson and O. A. Gross. Incidence matrices and interval graphs. *Pacific Journal of Mathematics*, 15(3):835 – 855, 1965.
- [FLRS11] F. V. Fomin, D. Lokshtanov, V. Raman, and S. Saurabh. Bidimensionality and EPTAS. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, San Francisco, California, USA, January 23-25, 2011*, pages 748–759, 2011.
- [Fly17] M. Flynn. The largest bond in 3-connected graphs. Honors Theses. 695, University of Mississippi, 2017. https://egrove.olemiss.edu/hon_thesis/695/.
- [FMU92] J. Fonlupt, A.R. Mahjoub, and J.P. Uhry. Compositions in the bipartite subgraph polytope. *Discrete Mathematics*, 105(1):73 – 91, 1992.
- [Gan13] A. Ganguly. Properties of cut polytopes. *University of Minnesota Digital Conservancy*, 2013.
- [GHK⁺18] R. Gandhi, M. T. Hajiaghayi, G. Kortsarz, M. Purohit, and K. K. Sarpatwar. On maximum leaf trees and connections to connected maximum cut problems. *Information Processing Letters*, 129:31–34, 2018.
- [GJ83] M.R. Garey and D.S. Johnson. Crossing number is NP-complete. *SIAM Journal on Algebraic Discrete Methods*, 4(3):312–316, 1983.
- [GKL⁺19] V. Grimm, T. Kleinert, F. Liers, M. Schmidt, and G. Zöttl. Optimal price zones of electricity markets: a mixed-integer multilevel model and global solution approaches. *Optimization Methods and Software*, 34(2):406–436, 2019.
- [GLV01] A. Galluccio, M. Loeb, and J. Vondrak. Optimization via enumeration: a new algorithm for the max cut problem. *Mathematical Programming*, 90:273–290, 2001.
- [GNS06] D. Golovin, V. Nagarajan, and M. Singh. Approximating the k -multicut problem. In *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2006, Miami, Florida, USA, January 22-26, 2006*, pages 621–630. ACM Press, 2006.
- [GP81] M. Grötschel and W.R. Pulleyblank. Weakly bipartite graphs and the max-cut problem. *Operations Research Letters*, 1(1):23–27, 1981.
- [Gro04] Martin Grohe. Computing crossing numbers in quadratic time. *Journal of Computer and System Sciences*, 68(2):285–302, 2004.

- [GVY06] N. Garg, V. Vazirani, and M. Yannakakis. Primal-dual approximation algorithms for integral flow and multicut in trees. *Algorithmica*, 18:3–20, 2006.
- [GW95] M.X. Goemans and D.P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42(6):1115–1145, 1995.
- [Had75] F.O. Hadlock. Finding a maximum cut of a planar graph in polynomial time. *SIAM Journal on Computing*, 4(3):221–225, 1975.
- [HKM⁺15] M. T. Hajiaghayi, G. Kortsarz, R. MacDavid, M. Purohit, and K. K. Sarpatwar. Approximation algorithms for connected maximum cut and related problems. In *Proc. of European Symposium on Algorithms, ESA 2015*, volume 9294 of *LNCS*, pages 693–704. Springer, 2015.
- [HKM⁺20] M. T. Hajiaghayi, G. Kortsarz, R. MacDavid, M. Purohit, and K. K. Sarpatwar. Approximation algorithms for connected maximum cut and related problems. *Theoretical Computer Science*, 814:74–85, 2020.
- [HLA17] A. Hornáková, J. H. Lange, and B. Andres. Analysis and optimization of graph decompositions by lifted multicuts. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1539–1548. PMLR, 2017.
- [HT73] J. Hopcroft and R. Tarjan. Dividing a graph into triconnected components. *SIAM Journal on Computing*, 2(3):135–158, 1973.
- [HT16] I. Heller and C. B. Tompkins. *14 . An Extension of a Theorem of Dantzig's*, pages 247–254. Princeton University Press, 2016.
- [HV91] D. J. Haglin and S. M. Venkatesan. Approximation and intractability results for the maximum cut problem and its variants. *IEEE Transactions on Computers*, 40(1):110–113, 1991.
- [HW07] J.M. Hochstein and K. Weihe. Maximum s - t -flow with k crossings in $O(k^3 n \log n)$ time. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2007, New Orleans, Louisiana, USA, January 7-9, 2007*, pages 843–847, 2007.
- [Kam12] M. Kamiński. MAX-CUT and containment relations in graphs. *Theoretical Computer Science*, 438:89–95, 2012.
- [Kar72] R. M. Karp. Reducibility among combinatorial problems. In *Proc. of a symposium on the Complexity of Computer Computations, ICCO*, pages 85–103, 1972.
- [Kar01] D.R. Karger. A randomized fully polynomial time approximation scheme for the all-terminal network reliability problem. *SIAM Review*, 43(3):499–522, 2001.
- [Kaw15] K. Kawarabayashi. Graph isomorphism for bounded genus graphs in linear time. *arXiv*, abs/1511.02460, 2015.
- [KKMT19a] Y. Kobayashi, Y. Kobayashi, S. Miyazaki, and S. Tamaki. An FPT algorithm for Max-Cut parameterized by crossing number. *arXiv*, abs/1904.05011, 2019.

- [KKMT19b] Y. Kobayashi, Y. Kobayashi, S. Miyazaki, and S. Tamaki. An improved fixed-parameter algorithm for Max-Cut parameterized by crossing number. In *Combinatorial Algorithms - 30th International Workshop, IWOCA 2019, Pisa, Italy, July 23-25, 2019, Proceedings*, pages 327–338, 2019.
- [KR07] K. Kawarabayashi and B.A. Reed. Computing crossing number in linear time. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing, San Diego, California, USA, June 11-13, 2007*, pages 382–390, 2007.
- [KR21] M. Koley and T. Römer. Seminormality, canonical modules, and regularity of cut polytopes. *Journal of Pure and Applied Algebra*, 226:106797, 05 2021.
- [Kra91] J. Kratochvíl. String graphs. II. recognizing string graphs is NP-hard. *Journal of Combinatorial Theory, Series B*, 52(1):67 – 78, 1991.
- [Kur30] C. Kuratowski. Sur le problème des courbes gauches en topologie. *Fundamenta Mathematicae*, 15(1):271–283, 1930.
- [KW00] V. Kaibel and M. Wolff. Simple 0/1-polytopes. *European Journal of Combinatorics*, 21(1):139 – 144, 2000.
- [LA20] J. H. Lange and B. Andres. On the lifted multicut polytope for trees. In *Pattern Recognition - 42nd DAGM German Conference, DAGM GCP 2020, Tübingen, Germany, September 28 - October 1, 2020, Proceedings*, volume 12544 of *Lecture Notes in Computer Science*, pages 360–372. Springer, 2020.
- [LM20] M. Lasoń and M. Michałek. A note on seminormality of cut polytopes. *arXiv*, abs/2012.07907, 2020.
- [LP12] F. Liers and G. Pardella. Partitioning planar graphs: A fast combinatorial approach for max-cut. *Computational Optimization and Applications*, 51(1):323–344, 01 2012.
- [Moh99] B. Mohar. A linear time algorithm for embedding graphs in an arbitrary surface. *SIAM Journal on Discrete Mathematics*, 12:6–26, 1999.
- [MR95] S. Mahajan and H. Ramesh. Derandomizing semidefinite programming based approximation algorithms. *SIAM Journal on Computing*, 28(5):1641–1663, 1995.
- [NW88] G.L. Nemhauser and L.A. Wolsey. *Integer and Combinatorial Optimization*. Wiley interscience series in discrete mathematics and optimization. Wiley, 1988.
- [OD72] G.I. Orlova and Y.G. Dorfman. Finding the maximal cut in a graph. *Cybernetics*, 10:502–504, 1972.
- [Ohs10] H. Ohsugi. Normality of cut polytopes of graphs is a minor closed property. *Discrete Mathematics*, 310:1160–1166, 2010.
- [Ohs14] H. Ohsugi. Gorenstein cut polytopes. *European Journal of Combinatorics*, 38:122–129, 2014.
- [PQ80] J.C. Picard and M. Queyranne. On the structure of all minimum cuts in a network and applications. In *Combinatorial Optimization II*, pages 8–16. Springer Berlin Heidelberg, Berlin, Heidelberg, 1980.
- [PT97] J. Pach and G. Tóth. Graphs drawn with few crossings per edge. *Combinatorica*, 17:427–439, 1997.

- [PY91] C.H. Papadimitriou and M. Yannakakis. Optimization, approximation, and complexity classes. *Journal of Computer and System Sciences*, 43(3):425–440, 1991.
- [RS95] N. Robertson and P. D. Seymour. Graph minors .XIII. the disjoint paths problem. *Journal of Combinatorial Theory, Series B*, 63(1):65–110, 1995.
- [RS18] T. Römer and S. Saeedi Madani. Retracts and algebraic properties of cut algebras. *European Journal of Combinatorics*, 69:214–236, 2018.
- [SS08] B. Sturmfels and S. Sullivant. Toric geometry of cuts and splits. *Michigan Math. J.*, 57:689–709, 2008.
- [SW10] M. Skutella and A. Weber. On the dominant of the s - t -cut polytope: Vertices, facets, and adjacency. *Mathematical Programming*, 124(1-2):441–454, 2010.
- [SWK90] W.K. Shih, S. Wu, and Y. S. Kuo. Unifying maximum cut and minimum cut of a planar graph. *IEEE Transactions on Computers*, 39(5):694–697, May 1990.
- [Tho99] R. Thomas. Recent excluded minor theorems for graphs. In *Survey in Combinatorics*, pages 201–222. Univ. Press, 1999.
- [Tut66] W. T. Tutte. *Connectivity in Graphs*. Univ. of Toronto Press, 1966.
- [VKR08] S. Vicente, V. Kolmogorov, and C. Rother. Graph cut based image segmentation with connectivity priors. In *Proc. of 2008 IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 1–8, 2008.
- [Wag37] K. Wagner. Über eine Eigenschaft der ebenen Komplexe. *Mathematische Annalen*, 114(1):570–590, 1937.
- [Wag60] K. Wagner. Bemerkungen zu Hadwigers Vermutung. *Mathematische Annalen*, 141(5):433–451, 1960.
- [YKCP83] M. Yannakakis, P. C. Kanellakis, S. S. Cosmadakis, and C. H. Papadimitriou. Cutting and partitioning a graph after a fixed pattern. In *Automata, Languages and Programming*, pages 712–722, Berlin, Heidelberg, 1983. Springer Berlin Heidelberg.
- [Zie00] G. M. Ziegler. Lectures on 0/1-polytopes. In *Polytopes—combinatorics and computation (Oberwolfach, 1997)*, volume 29 of *DMV Sem.*, pages 1–41. Birkhäuser, Basel, 2000.
- [Zie12] G. M. Ziegler. *Lectures on Polytopes*. Graduate Texts in Mathematics. Springer New York, 2012.