Multicut Optimization Guarantees & Geometry of Lifted Multicuts

A dissertation submitted towards the degree
Doctor of Natural Sciences
of the Faculty of Mathematics and Computer Science
of Saarland University

by

Jan-Hendrik Lange, M.Sc.

Saarbrücken
2020

Day of Colloquium	3rd of July, 2020
Dean of the Faculty	Univ.-Prof. Dr. Thomas Schuster

Examination Committee

Chair	Prof. Dr. Kurt Mehlhorn
Reviewer, Advisor	Prof. Dr. Björn Andres
Reviewer	Prof. Dr. Bernt Schiele
Reviewer	Prof. Dr. Tomáš Werner
Reviewer	Prof. Dr. Thomas Pock
Academic Assistant	Dr. Paul Swoboda

ABSTRACT

Clustering of graph structured data is an important task across many areas of modern data science such as image analysis and machine learning. A particular model for partitioning graphs purely based on similarities and dissimilarities defined on the edges is known as the multicut problem in combinatorial optimization. Its key feature is that the number of components of the partition need not be specified, but is determined as part of the optimization process given the edge information. This promotes its application to various problems in image segmentation, social network analysis and computer vision. There are two challenges associated with the NP-hard multicut problem as a model for large scale graph partitioning, which we adress in this thesis.

Firstly, the methods that are commonly employed to solve practical multicut problems are fast heuristics, due to the large size of the instances. Therefore, the provided solutions come without any (non-trivial) guarantees, neither in terms of the objective value nor in terms of the variable values. In this thesis we develop methods to provide such guarantees in regimes where common linear programming methods are intractable. Our algorithms allow to compute non-trivial lower bounds as well as partial variable assignments of optimal solutions efficiently for large instances. We demonstrate the effectiveness of our methods on benchmark data sets as well as instances from biomedical imaging. The good performance of our approach is facilitated by the inherent structural simplicity that practical instances often exhibit. Towards a further theoretical explanation for this phenomenon, we study sign patterns of the cost vector that prohibit tightness of the standard linear programming relaxation of the multicut problem. Our results enhance the understanding of the practical complexity of signed graph partitioning.

Secondly, multicuts do not encode the same-cluster relationship for pairs of nodes that are not adjacent in the underlying graph. A generalization of the multicut problem that allows to model such inter-cluster connectivity for arbitrary pairs of nodes is known as the lifted multicut problem. However, integer linear optimization for this higher-order version of the problem becomes more challenging and requires an analysis of the (convex hull of its) feasible set, the lifted multicut polytope. We study the polyhedral geometry of lifted multicuts by investigating under which conditions the fundamental inequalities associated with lifted multicuts define facets of the lifted multicut polytope. For the special case of trees, we draw a connection to pseudo-Boolean optimization and give a tighter description of the lifted multicut polytope, which is exact when the tree is a simple path. For the moral lineage tracing problem, which is a closely related clustering formulation for joint segmentation and tracking of biological cells in images, we present a tighter description of the associated polytope and an improved branch-and-cut method. Our results are valuable from a theoretical and practical perspective as they enable faster algorithms to solve the lifted multicut problem and related variants.

In essence, this thesis develops methods and analyses for multicut optimization guarantees and investigates the polyhedral geometry of lifted multicuts.

ZUSAMMENFASSUNG

Die Partitionierung von Daten mit Graphenstruktur ist eine wichtige Aufgabe in vielen Bereichen der modernen Datenwissenschaften wie der Bildanalyse und dem maschinellen Lernen. Ein bestimmtes Modell zur Graphenpartitionierung, welches alleine auf den Kanten zugeordneten Ähnlichkeits- und Unähnlichkeitswerten basiert, ist unter dem Namen Mehrfachschnittproblem in der kombinatorischen Optimierung bekannt. Seine Schlüsseleigenschaft besteht darin, dass die Anzahl der Komponenten der Partition nicht spezifiziert werden muss, sondern als Teil des Optimierungsprozesses aus den Kanteninformationen bestimmt wird. Dies fördert seine Anwendung auf diverse Probleme in der Bildsegmentierung, sozialen Netzwerkanalyse und dem maschinellen Sehen. In dieser Arbeit behandeln wir die folgenden zwei Herausforderungen im Zusammenhang mit dem NP-schweren Mehrfachschnittproblem als Modell zur Partitionierung von sehr großen Graphen.

Erstens, aufgrund der Größe der Instanzen, werden gebräuchlicherweise schnelle Heuristiken angewandt, um praktische Mehrfachschnittprobleme zu lösen. Daher fehlen den berechneten Lösungen jegliche (nicht-triviale) Garantien, sowohl was den Zielfunktionswert als auch die Werte der Variablen betrifft. In dieser Arbeit entwickeln wir Methoden um dann solche Garantien bereitzustellen, wenn gewöhnliche Methoden der linearen Programmierung zu aufwendig sind. Unsere Algorithmen erlauben es sowohl nicht-triviale untere Schranken als auch partielle Variablenbelegungen von optimalen Lösungen für große Instanzen effizient zu berechnen. Wir demonstrieren die Effektivität unserer Methoden auf Vergleichsdatensätzen und Instanzen der biomedizinischen Bildverarbeitung. Die gute Leistungsfähigkeit unseres Ansatzes wird dadurch gefördert, dass praktische Instanzen oftmals eine inhärente strukturelle Einfachheit aufweisen. In Richtung einer tieferen theoretischen Erklärung dieses Phänomens studieren wir die Vorzeichenmuster der Zielfunktion, die eine exakte Lösung des Mehrfachschnittproblems mit Hilfe der linearen Relaxierung verhindern. Unsere Resultate erweitern das Verständnis der praktischen Komplexität der Partitionierung von signierten Graphen.

Zweitens kodieren Mehrfachschnitte nicht direkt, ob nicht-benachbarte Knoten des zugrundeliegenden Graphen zur gleichen Komponente der Partition gehören. Eine Verallgemeinerung des Mehrfachschnittproblems, welche es erlaubt diese Verbundenheit innerhalb der Komponenten für beliebige Paare von Knoten zu modellieren, wird hochgezogenes Mehrfachschnittproblem genannt. Die ganzzahlige lineare Optimierung für diese Version des Problems mit höherer Ordnung wird jedoch herausfordernder und verlangt eine Analyse der (konvexen Hülle der) zulässigen Menge, dem hochgezogenen Mehrfachschnitt-Polytop. Wir studieren die polyhedrale Geometrie von hochgezogenen Mehrfachschnitten indem wir untersuchen, unter welchen Bedingungen die mit hochgezogenen Mehrfachschnitten assozierten fundamentalen Ungleichungen facettendefinierend für das hochgezogene Mehrfachschnitt-Polytop sind. Für den speziellen Fall von Bäumen zeigen wir eine Verbindung zu pseudo-boolescher Optimierung auf und geben eine engere Beschreibung des hochgezogenen Mehrfachschnitt-Polytops an, die im Falle eines einfachen Pfades exakt

ist. Für das moralische Abstammungsverfolgungs-Problem, welches eine eng verwandte Formulierung zur gemeinsamen Segmentierung und Verfolgung von biologischen Zellen in Bilddaten darstellt, präsentieren wir eine engere Beschreibung des assozierten Polytops und eine verbesserte Branch-and-Cut-Methode. Unsere Resultate sind aus theoretischer und praktischer Sicht wertvoll, da sie schnellere Algorithmen zur Lösung von hochgezogenen Mehrfachschnittproblemen und verwandten Formulierungen ermöglichen.

Kurzum entwickelt diese Arbeit Methoden und Analysen für Garantien in der Mehrfachschnittoptimierung und untersucht die polyhedrale Geometrie von hochgezogenen Mehrfachschnitten.

CONTENTS

1 INTRODUCTION

In modern data science graphs are ubiquitous as models for structured data. Graphs are applicable whenever the structure of the data can be broken down into a number of pairwise relationships, an assumption that spans a wide range of applications. Important examples are physical road networks, computer networks, social networks and digital images. For this reason, vast amounts of research have been devoted to the study of graphs and optimization problems on graphs. Besides the classical problems that can be solved in polynomial time such as shortest paths, matchings and maximum flows, there is considerable interest in computationally hard problems. This thesis is concerned with the NP-hard multicut problem, which models graph partitioning tasks purely based on edge information. Informally, it seeks a partition of the graph which minimizes the sum of real-valued costs associated with the edges that straddle distinct components.

Partitioning (or clustering) is an important problem that arises in many different contexts naturally when one seeks to find meaningful groups within the data. Examples include the detection of communities in social networks and the extraction of neural structures from electron microscopy image data of the brain.

In this thesis we study the multicut problem and variants thereof from an optimization perspective. Our contributions range from algorithms that help to identify optimal solutions for large-scale instances of the problem in practice to a theoretical analysis of the polyhedral geometry of the problem.

In Section 1.1 we first summarize the contributions of this thesis and provide brief descriptions of the remaining chapters. Then, in Section 1.2, we introduce the multicut problem mathematically and review the related work. The section provides both the necessary technical background and motivation for the main part of this thesis.

1.1 CONTRIBUTIONS AND OUTLINE

The contributions of this thesis can be divided into two parts.

The first part is concerned with practical and theoretical guarantees for multicut optimization. Since practitioners want to solve large instances of the problem, they commonly apply primal heuristics. A drawback of this approach is that it does not give any assertions about the quality of the solution. However, it is observed that, although very large, practical instances often are structurally rather simple. This is indicated by high quality heuristic solutions and fairly tight linear programming relaxations (on small instances), often due to the very accurate estimation of the cost vector. In our contributions we provide insight as to what structural properties distinguish hard from simple instances and develop algorithmic methods to exploit simple substructures. Here, we address three different issues.

Firstly, we propose to scale the application of linear programming bounds for the multicut problem to much larger size regimes by computing dual solutions heuristically with a fast combinatorial algorithm. Our approach enables the computation of nontrivial lower bounds for instances that are too large for more sophisticated linear programming methods.

Secondly, we provide local conditions on the cost structure of the problem that allow to identify partial variable assignments that are optimal. Moreover, we develop algorithmic techniques to check the conditions efficiently. Our method allows to compute parts of optimal solutions for large instances even when exact optimization is intractable.

Thirdly, we provide theoretical results towards the characterization of cost vectors for which the cycle relaxation is tight. We identify several sign patterns that prohibit tightness of the cycle relaxation in general. Our results enhance the understanding of the hardness of the problem and the good performance of linear programming relaxations in practice.

The second part of this thesis is concerned with the polyhedral geometry of a higher-order variant of the multicut problem, known as the lifted multicut problem. In contrast to the multicut problem, the lifted multicut problem enables to assign a cost to non-neighboring pairs of nodes in the graph. Therefore, it is a more expressive model for segmentation and clustering applications. To solve an integer linear programming formulation of the lifted multicut problem by branch-and-cut, an analysis of its feasible set, the lifted multicut polytope, is critical. Therefore, we study the geometry of lifted multicuts in this part of the thesis. Our results are separated into three chapters, which treat different variants of the problem.

Firstly, we provide an analysis of the lifted multicut polytope in the most general case. We analyse the fundamental inequalities associated with the lifted multicut polytope and determine conditions on when they are facet-defining. For box inequalities, we give necessary and sometimes sufficient conditions. For cycle inequalities, we give a characterization of the facet-defining property. For cut inequalities, we establish a number of necessary conditions. Our results generalize some of the prior work on the multicut polytope.

Secondly, we study the lifted multicut polytope for the special case of trees. Then, the lifted multicut problem reduces to the minimization of a sparse multi-linear polynomial over binary variables. We provide a tighter description of the lifted multicut polytope in this case, which is exact when the tree is a simple path. Our results establish a connection between lifted multicuts and pseudo-Boolean optimization and extend the polyhedral analysis to the extreme case of trees, which are the minimally connected graphs.

Thirdly, we study the geometry associated with the moral lineage tracing problem, which is a clustering formulation for the joint segmentation and tracking of cells and closely related to lifted multicuts. We analyse the feasible set of the problem and thereby provide a tighter description in terms of linear inequalities. Our results yield a branch-and-cut algorithm that exhibits improved performance upon prior work.

Apart from the introduction, the thesis is organized into seven additional chapters, which we summarize as follows.

Chapter 2: Heuristic Lower Bounds Most of the work on solving large scale multicut instances has been devoted to developing fast primal heuristics, which often return high quality solutions in practical scenarios. From an optimization perspective, however, these algorithms are unsatisfactory, as they provide no bound on the distance from the optimal solution. In order to obtain lower bounds one has to resort to methods that build upon linear programming relaxations, which do not scale very well for large problem sizes. In this chapter we explore a fast heuristic approach to compute solutions of a dual linear program associated with the multicut problem. We show that we can compute valuable dual information for practical instances fast. The chapter is based on (Lange et al., 2018) from which we present the results concerning the dual heuristic algorithm.

Chapter 3: Partial Optimality Besides missing lower bounds on the objective value, primal heuristics generally provide no insight whether parts of solution they compute are in fact optimal. Even linear programming relaxations are in general unable to determine which variable assignments are optimal, as long as the relaxation is not tight. In this chapter we derive local conditions on the cost structure of the problem that allow us to compute optimal assignments for a subset of the variables. Moreover, we develop fast combinatorial routines that can check our conditions efficiently, which allows us to shrink the problem sizes of practical instances. In numerical experiments on benchmark data sets as well as large biomedical instances from practice we demonstrate the effectiveness of our method. Besides the multicut problem, we also treat the max-cut problem in a joint manner. The chapter is based on the partial optimality results from (Lange et al., 2018, 2019).

Chapter 4: Tightness of the Cycle Relaxation In practice it is often observed that linear programming relaxations are fairly tight, which means they exhibit small optimality gaps and many variables in the relaxed solution are integer. The structural reasons for this phenomenon remain unclear, however. In this chapter we aim to identify classes of signed graphs whose multicut instances can be solved exactly via the cycle relaxation for all cost vectors whose sign pattern agrees with the graph signature. Our approach is inspired by the analogous question for the max-cut problem and leads to the study of integrality of covering polyhedra associated with signed graphs, which can be characterized in terms of forbidden minors. We provide an exact characterization for two special classes of graphs and further necessary conditions for general graphs. The chapter is based on the work presented in (Lange, 2020).

Chapter 5: Lifted Multicuts As multicuts only encode the same-cluster relationship between neighboring pairs of nodes, they are blind to the differences in graph partitions that induce the same multicut. Moreover, the multicut problem is unable to model interactions between nodes in the graph that are far apart. Considering partitions into connected components only, we introduce a lifting of multicuts so as to encode the same-cluster relationship for arbitrary pairs of nodes. We study the geometry of the lifted multicut polytope, the feasible set of the lifted multicut problem. We give necessary and sometimes sufficient conditions on the facet-inducing property of the fundamental inequalities associated with the lifted multicut polytope. The chapter is based on the polyhedral results that constitute one of two equal contributions to (Horňáková et al., 2017). In our presentation, we correct some minor technical flaws of the proofs given therein, which does not alter the results.

Chapter 6: Tree Partitions In this chapter we study the partitioning of trees based on pairwise similarities and dissimilarities. The problem is formulated either as the minimization of a sparse multi-linear polynomial over binary variables or, equivalently, as a lifted multicut problem on a tree. We study the lifted multicut polytope for this special case and provide a tighter relaxation compared to the standard relaxation. We characterize several classes of facets and show that our inequalities yield a complete description of the lifted multicut polytope for the special case of a simple path. The chapter is based on the material from (Lange and Andres, 2017).

Chapter 7: Moral Lineage Tracing Simultaneous segmentation and tracking of cells in biomedical image data has been formulated by Jug et al. (2016) as a clustering problem on a hypothesis graph. The resulting problem, called moral lineage tracing, is closely related to the lifted multicut problem and can be solved by a branch-and-cut algorithm. In this chapter we improve upon the optimization method of Jug et al. (2016) by strengthening the integer linear programming formulation that they employ. Our insights lead to a faster algorithm, which we demonstrate on their original instances and new larger problems. The chapter is based on the integer linear programming part from (Rempfler et al., 2017). The paper has two major equal contributions, namely our improvements of the branch-and-cut algorithm and the primal heuristics for the problem contributed by Markus Rempfler. Note that the contents of the chapter appear similarly in the PhD thesis (Rempfler, 2019).

Chapter 8: Conclusion We conclude this thesis with the final chapter, which puts our results into context with each other. Firstly, we observe that our results improve the understanding of the practical complexity of the multicut problem. It can be judged by primal-dual gaps on large instances in practice, the extent to which large parts of optimal solutions can be computed efficiently, and sign patterns of the objective function that obstruct tightness of the cycle relaxation. Secondly, our polyhedral analysis of lifted multicut problems helps to understand and alleviate the additional complexity that is introduced to branch-and-cut algorithms by variables for non-neighboring pairs of nodes. Furthermore, we elaborate on possible avenues for future research. These include, for instance, stronger partial optimality criteria and novel cutting planes for the lifted multicut problem.

1.2 THE MULTICUT PROBLEM

In this section we introduce the multicut problem formally and summarize the related work as well as applications. It serves as a technical introduction and motivates the remainder of this thesis.

1.2.1 Notation

In this thesis we employ the following notation and conventions. We consider graphs $G = (V, E)$, which we assume to be connected throughout the thesis. For pairs of nodes

$u, v \in V$ with $u \neq v$ we write $uv = \{u, v\} = vu$ for short. For any subsets $U, W \subseteq V$ we write

$$\delta(U, W) = \{uv \in E \mid u \in U, w \in W\}$$

for the edges between U and W and $\delta(U) = \delta(U, V \setminus U)$ for the cut induced by $U \subseteq V$. Similarly, we write

$$E(U) = \{uv \in E \mid u \in U, v \in U\}$$

for the edge set induced by $U \subseteq V$. A subgraph of G is a graph $H = (V_H, E_H)$ such that $V_H \subseteq V$ and $E_H \subseteq E$. We identify common subgraphs such as paths and cycles with their edge sets and thus write $e \in H$ instead of $e \in E_H$ if it is clear from the context. All cycles considered in this thesis are simple by default (i.e. they are *circuits*).

Any vector x is a column vector and $x^\top$ its transpose. For $x \in \mathbb{R}^E$ and $F \subseteq E$, we denote the subvector of x that corresponds to F by x_F. The characteristic vector $\mathbb{1}_F$ takes value 1 for any $e \in F$ and 0 otherwise. Furthermore, we write $\mathbb{R}_+$ for the nonnegative quadrant.

1.2.2 Problem Formulation in Analogy to Max-Cut

Let $G = (V, E)$ be a connected graph. A *partition* of V is a set

$$\Pi = \{U_1, \ldots, U_k\}$$

of subsets $U_i \subseteq V$, where k is arbitrary, such that $V = \bigcup_i U_i$ and $U_i \cap U_j = \emptyset$ for all $i \neq j$. We call Π a *graph partition* of G and the U_i its *components*. Alternatively, we may also refer to any graph partition as a *clustering* of the graph and call the components accordingly *clusters*. Every graph partition is associated with the set of edges within, respectively between the components of the partition. The latter set is also known as *multicut*.

Definition 1.1 (Multicut)**.** Let $\Pi = \{U_1, \ldots, U_k\}$ be any partition of a graph $G = (V, E)$. The set of edges

$$M = \bigcup_{1 \leq i < j \leq k} \delta(U_i, U_j)$$

is called *multicut* of G associated with Π. Clearly, every partition is associated with a unique multicut, the converse does not hold true, cf. Figure 1.1.

The following equivalent characterization of multicuts is due to Chopra and Rao (1993), which we prove for the sake of completeness.

Lemma 1.1 (Chopra and Rao (1993))**.** *A set $M \subseteq E$ is a multicut of G if, and only if, for every cycle C of G it holds that $|M \cap C| \neq 1$.*

Proof. Suppose M is a multicut of G induced by a graph partition Π. Any cycle C is either entirely contained in a single component $U \in \Pi$ or it crosses the boundaries between components at least twice.

Figure 1.1: Two different graph partitions of the same graph that induce the same multicut. Shaded regions of the same color correspond to components of the partition. The associated multicut is indicated by dotted edges.

Conversely, assume $M \subseteq E$ is such that for any cycle C of G it holds that $|M \cap C| \neq 1$. Take an arbitrary connected component $H = (V_H, E_H)$ of the graph $(V, E \setminus M)$. If V_H does not induce H, then there exists an edge $uv \in M$ such that $u \in V_H$ and $v \in V_H$. However, since H is connected, there must be a cycle containing uv that violates the assumption. Thus, any connected component is induced by its set of nodes, which gives rise to a partition whose multicut is M. □

Definition 1.2 (Multicut Polytope)**.** The convex hull of characteristic vectors of multicuts of G is called *multicut polytope* w.r.t. G and is denoted by

$$\mathsf{MC}(G) = \operatorname{conv}\left\{\mathbb{1}_M \mid M \text{ multicut of } G\right\}.$$

We write $\mathsf{MC} = \mathsf{MC}(G)$ for short whenever this is clear from the context.

Definition 1.3 (Multicut Problem)**.** Let $\theta \in \mathbb{R}^E$ be a vector associated with the edges of G. The *multicut problem* w.r.t. G and θ is to find a minimum multicut of G w.r.t. θ. More precisely, it is written in the form

$$\min_{x \in \mathsf{MC}} \sum_{e \in E} \theta_e x_e \tag{P_{MC}}$$

as a binary optimization problem with a linear objective function. Note here that we take the liberty of identifying integral polyhedra with the set of their integer points, as the minimum of a linear program is attained at a vertex of the associated polytope.

For fixed $\theta \in \mathbb{R}^E$ we distinguish the *positive*, respectively *negative* edges

$$E^+ = \{e \in E \mid \theta_e > 0\} \quad \text{and} \quad E^- = \{e \in E \mid \theta_e < 0\}.$$

Note that any edge that is neither positive nor negative is irrelevant to the multicut problem and can thus be removed a-priori.

An important motiviation for the formulation (P_{MC}) of the multicut problem in our thesis is the analogy that it establishes with the max-cut problem. Classically, the max-cut problem is to find a cut of a graph $G = (V, E)$ that is maximal w.r.t. some nonnegative edge weights. Since this is equivalent to finding a minimum cut w.r.t. to the inverted weights, we can w.l.o.g. write it in a more general form that is analogous to (P_{MC}). To this end, let

$$\mathsf{CUT} = \operatorname{conv}\left\{\mathbb{1}_{\delta(U)} \mid U \subseteq V\right\} \subseteq \mathsf{MC}$$

denote the convex hull of characteristic vectors of cuts of G. The set CUT is called *cut polytope* of G and it is a subset of the multicut polytope, since every cut is also a multicut.

Definition 1.4 (Max-Cut)**.** Let $\theta \in \mathbb{R}^E$ be a vector associated with the edges of G. The *max-cut problem* w.r.t. G and θ is written in the form

$$\min_{x \in \mathsf{CUT}} \sum_{e \in E} \theta_e x_e \tag{$\mathrm{P}_{\mathsf{CUT}}$}$$

as a binary optimization problem with a linear objective function.

1.2.3 Integer Linear Program

In this section we review the description of the multicut polytope MC in terms of linear inequalities and integer constraints. The latter is achieved via the characterization given by Lemma 1.1. With any cycle C of G and any $f \in C$ we associate the so-called *cycle inequality*

$$x_f \le \sum_{e \in C \setminus \{f\}} x_e. \tag{1.1}$$

Definition 1.5 (Cycle Relaxation)**.** The *cycle relaxation* of the multicut polytope is denoted by

$$\mathsf{CYC} = \left\{x \in [0,1]^E \mid x \text{ satisfies (1.1) for all cycles } C \text{ and } f \in C\right\}.$$

The corresponding linear programming relaxation of (P_{MC}) can thus be written as

$$\min_{x \in \mathsf{CYC}} \sum_{e \in E} \theta_e x_e. \tag{$\mathrm{P}_{\mathsf{CYC}}$}$$

An integer linear programming formulation of (P_{MC}) is obtained by adding integer constraints to ($\mathrm{P}_{\mathsf{CYC}}$), according to the following lemma.

Lemma 1.2 (Chopra and Rao (1993))**.** *It holds that* $\mathsf{MC} = \operatorname{conv}\left(\mathsf{CYC} \cap \mathbb{Z}^E\right)$.

Cycle inequalities can be separated in polynomial time by shortest path methods such as Dijkstra's algorithm. A cycle inequality defines a facet of MC if, and only if, the associated cycle is simple and chordless (Chopra and Rao, 1993). Moreover, the non-simple and chordal cycles are in fact redundant in the description of CYC. Therefore, it suffices to separate the simple chordless cycles in order to solve ($\mathrm{P}_{\mathsf{CYC}}$), respectively ($\mathrm{P}_{\mathsf{MC}}$), with a cutting-plane approach.

Another important class of inequalities for the description of MC are associated with wheel subgraphs. A *wheel* is a graph $W = (V_W, C \cup E_u)$ with $V_W = \{v_1, \dots, v_{|C|}\} \cup \{u\}$ such that the subgraph $(\{v_1, \dots, v_{|C|}\}, C)$ is a cycle and

$$E_u = \{uv_i \mid 1 \le i \le k\}$$

is a set of edges that connect each node on the cycle to the distinguished node u. A wheel is called *odd* if $|C| = |E_u|$ is odd. For any wheel W that is a subgraph of G, we have the associated *wheel inequality*

$$\sum_{e\in C} x_e - \sum_{e\in E_u} x_e \le \left\lfloor \frac{|C|}{2} \right\rfloor. \tag{1.2}$$

Wheel inequalities are valid for MC and facet-defining if the associated wheel is odd (Chopra and Rao, 1993). Moreover, they can be separated in polynomial time.

Further polyhedral results are due to Grötschel and Wakabayashi (1989); Grötschel and Wakabayashi (1990); Deza et al. (1990, 1992); Chopra (1994).

1.2.4 Equivalent Formulations

The multicut problem appears under various names in the literature. In this section we gather common equivalent formulations and survey the related work that is connected with each of them.

Classical Formulation In mathematics, the multicut problem has been formulated early as a generalization of the classical min-cut problem to more than one terminal pair. Given nonnegative costs $c_e \ge 0$ for all $e \in E$ and k terminal pairs $(s_1, t_1), \dots, (s_k, t_k) \in V \times V$, where k is arbitrary, the classical formulation of the multicut problem is to find a minimum cost set of edges $M \subseteq E$ such that all terminal pairs are disconnected in the graph $(V, E \setminus M)$.

Lemma 1.3 (Demaine et al. (2006))**.** *The classical multicut problem is equivalent to* ($\mathrm{P_{MC}}$).

Proof. Consider an instance of the classical multicut problem induced by $G = (V, E)$, $c \ge 0$ and $(s_i, t_i) \in V \times V$ for $1 \le i \le k$. Define the augmented graph $G' = (V, E')$ where E' is obtained from E by adding an edge $\{s_i, t_i\}$ for every $1 \le i \le k$. Furthermore, define

$$\theta_e = \begin{cases} c_e & e \in E \\ -(\sum_{e\in E} c_e + 1) & e \in E' \setminus E. \end{cases}$$

Clearly, if x^* solves ($\mathrm{P_{MC}}$) w.r.t. G' and θ, then x^* restricted to E is a characteristic vector of an optimal solution for the classical formulation.

Conversely, consider an instance of ($\mathrm{P_{MC}}$) induced by $G = (V, E)$ and $\theta \in \mathbb{R}^E$. Enumerate the negative edges such that $E^- = \{u_1 v_1, \dots, u_k v_k\}$. Now, define an instance of the classical multicut problem w.r.t. $G' = (V', E')$ and $c \ge 0$ as follows. For each negative edge add another vertex to V such that $V' = V \cup \{w_1, \dots, w_k\}$. Then, construct E' from E by replacing each negative edge $u_i v_i \in E^-$ with an edge $u_i w_i$. Further, for each new vertex w_i we introduce a terminal pair (w_i, v_i) and define the cost vector $c \ge 0$ according to $c_e = \theta_e$ for all $e \in E^+$ and $c_{u_i w_i} = -\theta_{u_i v_i}$ for each $u_i v_i \in E^-$. Let M be an minimal cost set of edges that separates all terminal pairs. Then the vector $x^* \in \{0,1\}$ defined by $x_e = 1$ if $e \in E^+ \cap M$ and $x_{u_i v_i} = 1$ if $u_i w_i \notin M$ is an optimal solution to ($\mathrm{P_{MC}}$). □

We refer to Schrijver (2003) for a large body of work on the classical multicut problem and its dual problem, the multi-commodity flow problem.

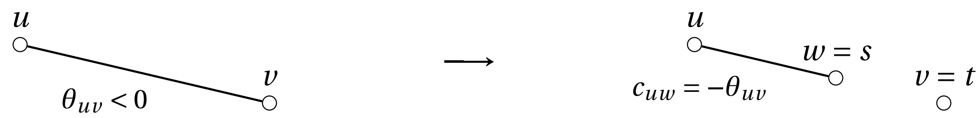

Figure 1.2: Replacing each negative edge by a positive edge and a terminal pair transforms the multicut problem (P_{MC}) to the classical formulation. Note that the transformation is not unique as the roles of u and v may be swapped.

Clique Partitioning If G is complete, then the multicut problem is also known as *clique partitioning* problem, which is defined as

$$\begin{aligned} \min \quad & \sum_{u \neq v} \theta_{uv} x_{uv} \\ \text{s.t.} \quad & x_{uv} \leq x_{uw} + x_{wv} \quad \forall u, v, w \in V \\ & x_{uv} \in \{0, 1\}. \end{aligned}$$

The equivalence follows simply from the fact that triangles are the only chordless cycles in a complete graph. Since G is complete, there is a one-to-one correspondence between partitions of G and its multicuts. Moreover, the components of any partition are cliques, hence the name clique partitioning (Grötschel and Wakabayashi, 1989).

Correlation Clustering Given nonnegative costs $c \in \mathbb{R}_+^E$ and edges labeled either + or −, the weighted *weighted correlation clustering* problem is defined as

$$\min_{x \in \mathrm{MC}} \sum_{e \in E^-} c_e(1 - x_e) + \sum_{e \in E^+} c_e x_e, \tag{1.3}$$

where E^+ and E^- collect the edges labeled + and −, respectively. Clearly, problem (1.3) is equivalent to (P_{MC}) for $c = |\theta|$ via

$$\begin{aligned} & \min_{x \in \mathrm{MC}} \sum_{e \in E^-} c_e(1 - x_e) + \sum_{e \in E^+} c_e x_e \\ = & \min_{x \in \mathrm{MC}} \sum_{e \in E^-} |\theta_e| + \sum_{e \in E^+} |\theta_e| x_e - \sum_{e \in E^-} |\theta_e| x_e \\ = & \min_{x \in \mathrm{MC}} \sum_{e \in E} \theta_e x_e - \sum_{e \in E^-} \theta_e. \end{aligned}$$

The name correlation clustering was coined by Bansal et al. (2004), who considered unit edge costs on complete graphs, established NP-hardness of the problem and gave first approximation results. They similarly considered a maximization version of (1.3), where E^+ and E^- are swapped. The resulting problem is equivalent at the optimum but differs in its approximability. A lot of the work on correlation clustering in the machine learning community has been devoted to approximation guarantees for the problem (1.3). Note that, because of the constant offset in objective values, these results do not directly carry over to problem (P_{MC}). Demaine et al. (2006) drew the connection between the classical multicut formulation and correlation clustering in general graphs. They also established its APX-hardness and gave an $\mathcal{O}(\log|V|)$ approximation algorithm. Further hardness results and improved approximation algorithms for particular classes of graphs and/or edge costs are due to Charikar et al. (2005); Chawla et al. (2006, 2015); Ailon et al. (2008, 2012); Klein et al. (2015); Veldt et al. (2017). Pan et al. (2015) develop parallelized algorithms for very large correlation clustering instances.

Conflicted Cycle Covering We call a cycle of G *conflicted* if it contains precisely one negative edge.[1] Further, we denote by

$$\mathscr{C}^-(G,\theta) = \{C \text{ cycle of } G \mid |C \cap E^-| = 1\}$$

the set of conflicted cycles of G.

One can show that in order to solve either (P_{MC}) or (P_{CYC}) it suffices to consider only the cycle inequalities w.r.t. conflicted cycles, where the negative edge is on the left-hand side of (1.1). More precisely, consider the polytope

$$\mathsf{CYC}^- = \left\{x \in [0,1]^E \mid x \text{ satisfies (1.1) for all } C \in \mathscr{C}^-(G,\theta) \text{ and } f \in C \cap E^-\right\}$$

We have the following lemma, which is a stronger version of a lemma appearing in (Yarkony et al., 2015).

Lemma 1.4 (Lange et al. (2018)). *For any $\theta \in \mathbb{R}^E$ it holds that*

$$\min_{x \in \mathsf{CYC}} \theta^\top x = \min_{x \in \mathsf{CYC}^-} \theta^\top x \tag{1.4}$$

and

$$\min_{x \in \mathsf{MC}} \theta^\top x = \min_{x \in \mathbb{Z}^E \cap \mathsf{CYC}^-} \theta^\top x. \tag{1.5}$$

Proof. Let x^* be an optimal solution to the right-hand side of (1.4). We show that x^* satisfies all cycle inequalities (1.1) by contradiction. To this end, suppose there exists a cycle C and $f \in C$ such that

$$x_f^* > \sum_{e \in C \setminus \{f\}} x_e^*.$$

If any edge $g \in C \setminus \{f\}$ is negative, then increasing x_g^* would lower the objective. Since x^* is optimal, there must be a conflicted cycle C' with $g \in C'$ such that $x_g^* = \sum_{e \in C' \setminus \{g\}} x_e^*$. Note that this means $f \notin C'$. We write $C \triangle C'$ for the cycle obtained from the symmetric difference of C and C'. Apparently, the cycle $C \triangle C'$ has one negative edge less and $f \in C \triangle C'$. Therefore, by repeating the argument, we may w.l.o.g. assume that all edges in $C \setminus \{f\}$ are positive. Now assume that f is positive as well, then decreasing x_f^* would lower the objective. Therefore, since x^* is optimal, there is a conflicted cycle C' with $f \in C'$ and $g \in C' \cap E^-$ such that

$$\begin{aligned} x_g^* &= x_f^* + \sum_{e \in C' \setminus \{f,g\}} x_e^* \\ &> \sum_{e \in C \setminus \{f\}} x_e^* + \sum_{e \in C' \setminus \{f,g\}} x_e^* \\ &\geq \sum_{e \in C \triangle C' \setminus \{g\}} x_e^*. \end{aligned}$$

[1] Other authors used terms such as “bad” or “erroneous” (Bansal et al., 2004; Demaine et al., 2006).

Note that $C \triangle C'$ is a conflicted cycle. Thus, we conclude that x^* violates a conflicted cycle inequality and hence cannot be feasible. This concludes the proof of (1.4), the argument for (1.5) is analogous. □

From this we can easily derive the conflicted cycle covering formulation.

Lemma 1.5. *The multicut problem* (P_{MC}) *is equivalent to the integer linear program*

$$\begin{aligned} \min \quad & \sum_{e \in E} |\theta| \hat{x}_e & & \\ s.t. \quad & \sum_{e \in C} \hat{x}_e \geq 1 & & \forall C \in \mathscr{C}^-(G,\theta) \\ & \hat{x}_e \geq 0 & & \forall e \in E \\ & \hat{x}_e \in \mathbb{Z} & & \forall e \in E \end{aligned} \tag{1.6}$$

and the LP relaxation (P_{CYC}) *is equivalent to the corresponding linear program. The objective values differ by the constant* trivial lower bound $L^{\text{triv}} = \sum_{e \in E-} \theta_e$.

Proof. We use Lemma 1.4. Define $\hat{x}$ via $\hat{x}_e = x_e$ for any positive edge $e \in E^+$ and $\hat{x}_e = 1 - x_e$ for any negative edge $e \in E^-$. Since any conflicted cycle $C \in \mathscr{C}^-(G,\theta)$ has precisely one repulsive edge, all conflicted cycle inequalities become covering inequalities. □

The conflicted cycle covering ILP (1.6) is very similar to the standard ILP for the classical multicut formulation. The subtle difference is that the latter covers paths that are defined by terminal pairs, while the former covers cycles that are determined by a single negative edge. The formulation (1.6) shares the objective value with the correlation clustering problem (1.3), but differs in the formulation of the constraint set.

1.2.5 Solution Methods

Branch-and-Cut The multicut problem (P_{MC}) can be solved by a branch-and-cut algorithm, separating chordless cycle inequalities. Other inequalities that further tighten the linear relaxation and can be efficiently separated include *(odd) wheel* and *(odd) bicycle wheel* inequalities (Chopra and Rao, 1993). A common approach in practice is to separate violated cycle inequalities only at integer points found during branch-and-bound, since this enables the application of breadth-first-search instead of Dijkstra's algorithm. Together with the powerful generic cuts added by solvers such as CPLEX or Gurobi an effective method is obtained. In fact, for practical instances it is often faster to solve the ILP with this approach than solving the LP relaxation in a pure cutting-plane manner (Kappes et al., 2015).

Specialized LP Methods Yarkony et al. (2012) develop a column-generating method for an alternative dual formulation in the case of planar graphs. For general graphs, Yarkony et al. (2015) propose a dual decomposition that is solved via the subgradient method. The dual formulation by Swoboda and Andres (2017) gives rise to a block coordinate ascent algorithm, which is commonly referred to as *message passing*. We call the method that incorporates cycle constraints only, i.e. solves problem (P_{CYC}), message passing with cycles (MPC). In order to produce integer feasible solutions, all LP methods rely on some rounding scheme that may include a primal heuristic.

Greedy Edge Contraction In order to compute good feasible solutions quickly, greedy algorithms are commonly employed. A simple strategy is to start from the partition into single-node clusters and then iteratively merge clusters with highest positive costs associate with the edges between them. The resulting algorithm was proposed and coined *greedy additive edge contraction* (GAEC) by Keuper et al. (2015b); Levinkov et al. (2017). Related variants that are greedy towards a decrease in cost averaged over the number of nodes (respectively edges) associated with each contraction were suggested by Kardoost and Keuper (2019); Bailoni et al. (2019). The modified approaches lead to a more balanced growth of the components of the partition.

Local Search The *Cut, Glue & Cut* algorithm proposed by Beier et al. (2014) iteratively improves an initial solution by splitting and merging components. A generalization of the well-known Kernighan-Lin heuristic for graph partitioning was proposed by Keuper et al. (2015b) and coined *Kernighan-Lin with joins* (KLj). The algorithm iteratively improves an initial solution by series of local modifications such as swapping nodes between components and joining components. Beier et al. (2015) propose a *fusion move* algorithm that iteratively improves an initial solution by optimizing over the reduced set of variables where the current iterate and a generated proposal solution disagree. To handle even larger instances, Pape et al. (2017) suggest a way to assemble a global solution from solutions on smaller, overlapping parts of the input graph.

1.2.6 Applications

The multicut problem finds applications in a variety of different fields. Common tasks in social network analysis are, for instance, link classification and community detection (Cesa-Bianchi et al., 2012; Veldt et al., 2018). Recently, there has been particular interest in applying multicuts and related variants such as lifted multicuts (cf. Chapter 5) to image analysis and computer vision problems. The popularity of the multicut model in these areas is due to its key feature that the number of components is not constrained but rather is inferred from the optimization process. Thus, it serves as a clustering model that enables to estimate the number of clusters purely based on pairwise similarities and dissimilarities. Applications from image analysis and computer vision are a major motiviation for this thesis as they give rise to very high-dimensional problems instances that often exhibit structural simplicity due to the fairly accurate estimation of the cost vector. Multicut models for the task of image segmentation with closed contours are provided by Andres et al. (2011); Kappes et al. (2011); Kim et al. (2014); Keuper et al. (2015b); Kappes et al. (2016a); Beier et al. (2017). A probabilistic graph clustering method using multicuts is due to Kappes et al. (2016b). In the field of computer vision, notable applications include motion segmentation (Keuper et al., 2015a; Keuper, 2017), human pose estimation (Pishchulin et al., 2016; Insafutdinov et al., 2016), object and person tracking (Tang et al., 2015, 2017; Insafutdinov et al., 2017) and instance segmenation (Kirillov et al., 2017).

Probabilistic Model While the multicut problem is applicable regardless of the source of the cost vector, in many applications the costs originate from a probabilistic model (Andres et al., 2011; Kappes et al., 2011). More precisely, suppose $p_{uv} \in]0,1[$ for $uv \in E$ are independently estimated quantities that indicate the probability of assigning u and v to distinct clusters. Then the probability of any multicut M is given as

$$\prod_{e\in M} p_e \prod_{e\notin M} (1-p_e).$$

Finding the most probable multicut given the (commonly conditional) edge probabilities, amounts to solving the maximum a-posteriori (MAP) inference task

$$\begin{aligned}
&\underset{M}{\operatorname{argmax}} \quad \prod_{e\in M} p_e \prod_{e\notin M}(1-p_e) \\
=\;& \underset{x\in \mathrm{MC}\cap\mathbb{Z}^E}{\operatorname{argmax}} \quad \prod_{e\in E} p_e^{x_e}(1-p_e)^{1-x_e} \\
=\;& \underset{x\in \mathrm{MC}\cap\mathbb{Z}^E}{\operatorname{argmin}} \quad -\sum_{e\in E} x_e \log(p_e) + (1-x_e)\log(1-p_e) \\
=\;& \underset{x\in \mathrm{MC}}{\operatorname{argmin}} \quad \sum_{e\in E} x_e \log\tfrac{1-p_e}{p_e},
\end{aligned}$$

which is a multicut problem with costs $\theta_e = \frac{1-p_e}{p_e}$. Apparently, the closer p_e is to either 0 or 1, the greater is the absolute value $|\theta_e|$. Moreover, the sign of θ_e is determined by which side of $\frac{1}{2}$ the probability p_e is on. Therefore, the practical complexity of an instance that derives its costs from a probabilistic model is influenced critically by the estimation of the probabilities p_e, as the objective function is dominated by large values of $|\theta_e|$ and the number of conflicts is determined by the sign pattern of θ.

Part I

Optimization Guarantees for Multicut

HEURISTIC LOWER BOUNDS 2

2.1 INTRODUCTION AND RELATED WORK

THE multicut problem has found applications in fields such as social network analysis (Cesa-Bianchi et al., 2012; Veldt et al., 2018) and computer vision (Kappes et al., 2011; Keuper et al., 2015a; Insafutdinov et al., 2016; Beier et al., 2017; Tang et al., 2017). Although fast heuristics have been developed that compute high quality solutions to practical problems (Beier et al., 2014; Keuper et al., 2015b; Levinkov et al., 2017), it remains challenging for large instances to determine non-trivial bounds on the optimality gap. When instance-dependent bounds are concerned, the standard approach is to (approximately) solve an LP relaxation with either classical linear programming methods or specialized solvers (Yarkony et al., 2012, 2015; Swoboda and Andres, 2017). However, even the most sophisticated methods may not scale well enough to the very large instances commonly occuring in computer vision. In this regime, we propose to compute a solution of a particular dual formulation heuristically, by a fast combinatorial algorithm. Our method capitalizes on the simple structure of the packing dual of the multicut LP relaxation. We demonstrate empirically on a number of benchmark instances, that our method exhibits a runtime/accuracy trade-off that is different from prior work, as it computes non-trivial lower bounds rapidly. Nonetheless, the dual solutions provided by our method are often accurate enough to carry valuable information about the primal objective. When re-weighting the parameters of the problem with help of the reduced costs of the determined dual solution, we can show to obtain better feasible solutions by greedy heuristics.

It was noted by Bansal et al. (2004) that packing conflicted triangles gives rise to a lower bound for the correlation clustering objective in complete graphs. Demaine et al. (2006) hinted at the more general case of conflicted cycles but do not investigate it further. To the best of our knowledge, we are the first to study the practical relevance of an algorithm that packs conflicted cycles to compute lower bounds.

2.2 DUAL HEURISTIC ALGORITHM

For the presentation of our method we employ a simple dual linear program for the multicut LP relaxation. To this end, recall from Section 1.2.4 the LP relaxation of the conflicted cycle covering multicut formulation (up to the constant $L^{\text{triv}} = \sum_{e \in E^-} \theta_e$):

$$\begin{aligned} \min \quad & |\theta|^\top \hat{x} && && (2.1)\\ \text{s.t.} \quad & \sum_{e \in C} \hat{x}_e \geq 1 && \forall C \in \mathscr{C}^-(G, \theta) \\ & \hat{x}_e \geq 0 && \forall e \in E. \end{aligned}$$

The corresponding "packing dual" program reads

$$\begin{aligned} \max \quad & \mathbb{1}^\top \lambda && && (2.2)\\ \text{s.t.} \quad & \sum_{C: e \in C} \lambda_C \leq |\theta_e| && \forall e \in E && (2.3)\\ & \lambda_C \geq 0 && \forall C \in \mathscr{C}^-(G,\theta). \end{aligned}$$

A heuristic solution of (2.1), and thus a lower bound for (2.2) can be found efficiently by a simple coordinate ascent approach in the packing dual. The resulting procedure is summarized in Algorithm 1 that we call Iterative Cycle Packing (ICP). It works as follows: In each iteration, it chooses a conflicted cycle C and increases λ_C as much as possible, i.e. until it hits an upper bound c_e (initially $|\theta_e|$) for some edge $e \in C$. Afterwards, it decreases the costs c_e of all edges $e \in C$ by λ_C and removes all edges of zero weight. These steps are repeated until there are no conflicted cycles left. Starting from the trivial lower bound L^{triv}, the bound is increased by λ_C in each iteration. See Figure 2.1 for an illustration of the algorithm.

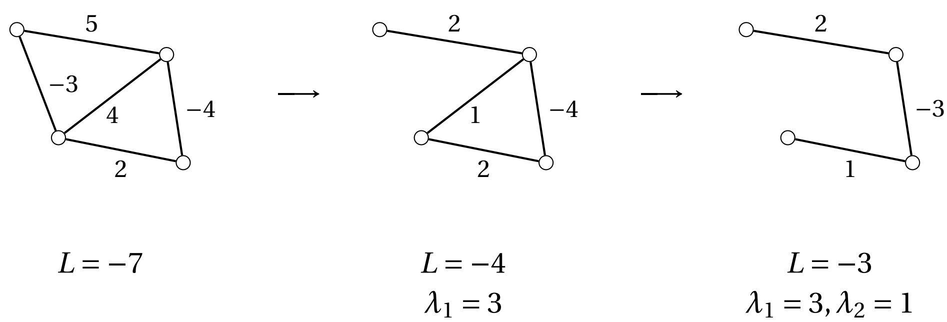

Figure 2.1: Illustration of the ICP algorithm. The method first increases λ_1 associated with the left conflicted triangle by 3. Then it increases λ_2 associated with the right conflicted triangle by 1.

Implementation Details The absolute running time of ICP as well as the quality of the output lower bounds depends on the choice of cycles C. We pursue the following strategy that we found to perform well empirically in both aspects: In each iteration of the main loop, we choose a negative edge $e = uv \in E^-$ such that u and v are in the same connected component of $G^+ = (V, E^+)$. Then, we find a conflicted cycle containing e by searching for a shortest path (in terms of hop distance) from u to v in G^+. We apply this search for conflicted cycles in rounds of increasing cycle length, using breadth-first search with an early termination criterion based on the hop distance. We also maintain and periodically update a component labeling of G^+ in order to to reduce the number of redundant shortest path searches.

2.2.1 Extension to Odd Wheels

In this section, we show that our dual heuristic algorithm can be extended so as to take into account odd wheel inequalities. In analogy to conflicted cycles, we call a wheel

Algorithm 1 Iterative Cycle Packing (ICP)

input $G = (V, E), \theta : E \to \mathbb{R}$
Initialize $c_e = |\theta_e|$ for all $e \in E$ and $\lambda = 0$, $L = L^{\text{triv}}$.
for $\ell = 3 \dots |E|$ **do**
  while $\exists C \in \mathscr{C}^-(G, \theta) : |C| \le \ell$ **do**
    Pick $C \in \mathscr{C}^-(G, \theta)$ such that $|C| \le \ell$.
    Compute $\lambda_C = \min_{e \in C} c_e$.
    Redefine $c_e = \begin{cases} c_e - \lambda_C & \text{if } e \in C \\ c_e & \text{else.} \end{cases}$
    Increase lower bound $L = L + \lambda_C$.
    Remove all edges $e \in E$ with $c_e = 0$ from G.
  end while
  if $\mathscr{C}^-(G, \theta) = \emptyset$ **then**
    return λ, L
  end if
end for

$W = (V_W, C \cup E_u)$ *conflicted* w.r.t. θ if $C \subset E^-$ and $E_u \subset E^+$. Now, for any conflicted wheel $W = (V_W, C \cup E_u)$, the definition $\hat{x}_e := 1 - x_e$ for all $e \in E^-$ transforms the associated wheel inequality (1.2) to the covering inequality

$$\sum_{e \in C \cup E_u} \hat{x}_e \ge \left\lceil \frac{|C|}{2} \right\rceil. \tag{2.4}$$

Therefore, the cycle covering relaxation (2.1) can be tightened by including the conflicted (odd) wheel inequalities (2.4). This correponds to introducing additional wheel variables λ_W with coefficients $\lceil |C|/2 \rceil$ into the dual (2.2). Note that in our heuristic approach we only increase dual variables, but never decrease any. Thus, since any conflicted odd wheel contains conflicted cycles, it only makes sense to increase wheel variables λ_W before the main loop of ICP (or in between). We search for conflicted odd wheels by, for each center vertex $u \in V$, finding an odd cycle in the positive neighbor subgraph $(\{v \neq u \mid uv \in E^+\}, E^-)$. Similar to the cycle case, we iteratively pack conflicted odd wheels until no more conflicted odd wheels are left. Afterwards, we enter the main loop of ICP.

2.2.2 Primal Re-weighting

A common approach in dual-based optimization for discrete graphical models is to reparameterize the primal ILP by means of a computed dual solution. This leaves the objective value of any primal solution unchanged, but may guide optimization algorithms to find better solutions (Swoboda and Andres, 2017).

Inspired by this approach we suggest to "re-weight" the costs of the primal LP (2.1) with the reduced costs of the heuristic dual solution obtained by ICP. The motivation is due to complementary slackness, which is made explicit in the following lemma.

Lemma 2.1. *Assume the optimal solution $\hat{x}^*$ of the primal LP (2.1) is binary, and the solution output by ICP solves the dual (2.2) optimally. Then, for every $e \in E$ with positive reduced cost $c_e > 0$, it holds that $\hat{x}^*_e = 0$.*

Proof. If $c_e > 0$, the constraint (2.3) at $e \in E$ is inactive at the optimal dual solution. Thus, $\hat{x}^*_e = 0$ in the optimal primal solution, by complementary slackness. □

Of course, the assumption of Lemma 2.1 is too strong for practical purposes. However, the intuition is that if the LP relaxation is fairly tight and the obtained dual solution is close to optimal, it can still provide useful information about the primal problem. More specifically, the costs c_e output by ICP can be interpreted as an indication of how likely the primal variable $\hat{x}_e$ is zero in an optimal solution. In order to make use of this information, we propose to shift the weights of the primal problem to a convex combination $\alpha|\theta_e| + (1-\alpha)c_e$ of the original and reduced costs, for a suitable choice of $\alpha \in (0,1)$, e.g. $\alpha = \frac{1}{2}$. This corresponds to changing the costs of the original problem (P_{MC}) to

$$\theta'_e = \alpha\theta_e + (1-\alpha)\left(|\theta_e| - \sum_{C:e\in C} \lambda_C\right)\operatorname{sign}\theta_e \qquad \forall e \in E.$$

Our experiments in Section 2.3 show that this shift can guide primal heuristics toward better feasible solutions to the original problem.

2.3 EXPERIMENTS

In this section we study the heuristic dual lower bounds and re-weightings empirically, for the instances from Kappes et al. (2015) and Leskovec et al. (2010).

Table 2.1: Reported below are lower bounds found by linear programming (LP), message passing (MPC), iterative cycle packing (ICP) and ICP with additional conflicted odd wheels, relative to the objective value of the best known feasible solution, as well as average total running times until convergence.

	Image Seg.		*Knott-3D-300*		*Knott-3D-450*		*Mod. Clust.*		*Epinions*		*Slashdot*	
	t [s]	Gap	t [s]	Gap	t [s]	Gap	t [s]	Gap	t [s]	Gap	t [s]	Gap
LP	1.31	0.03%	148.54	0.00%	2174.95	0.01%	0.98	5.59%	–	–	–	–
MPC	7.42	0.03%	17.34	0.00%	1236.28	0.01%	1.13	9.52%	36012	0.10%	7419	7.54%
ICP	0.10	0.21%	0.54	0.16%	3.31	0.18%	0.03	11.08%	65	1.04%	24	8.60%
OW+ICP	0.10	0.20%	0.77	0.17%	7.70	0.19%	0.03	9.51%	340	3.38%	63	8.51%

Instances From Kappes et al. (2015), we consider all three collections of instances: *Image Segmentation* contains instances w.r.t. planar superpixel adjacency graphs of photographs. *Knott-3D* contains instances w.r.t. non-planar supervoxel adjacency graphs of volume images taken by a serial sectioning electron microscope. *Modularity Clustering* contains instances w.r.t. complete graphs. In all three collections, the edge costs θ_e are fractional and non-uniform. For all these instances, except one in the collection *Modularity Clustering*, optimal solutions are accessible and are computed here as a reference. From Leskovec et al. (2010), we consider directed graphs of the social networks *Epinions* and *Slashdot*, each with more than half a million edges labeled either $+1$ or -1. Instances of the multicut problem are defined here by removing the orientation of edges, by deleting all self-loops, and by replacing parallel edges by a single edge with the sum of their costs[1].

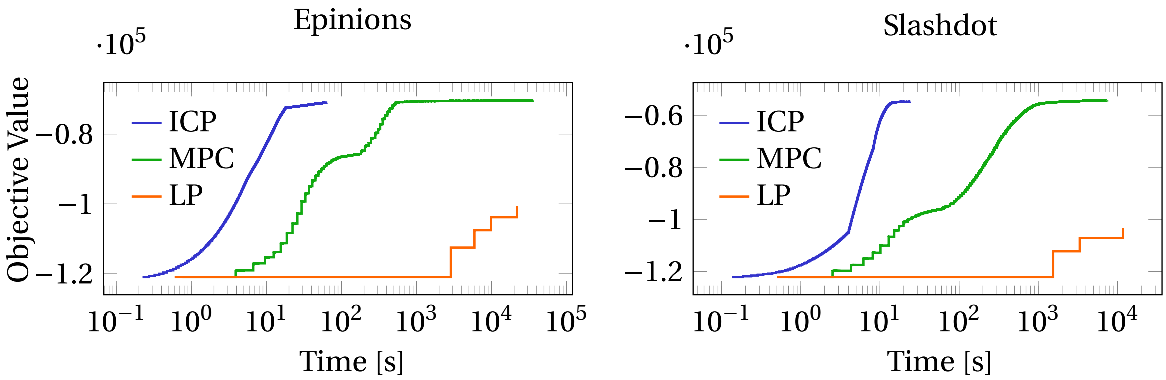

Figure 2.2: Shown above is the convergence of lower bounds found by linear programming (LP), message passing (MPC) and iterative cycle packing (ICP), for the instances *Epinions* and *Slashdot*.

In order to put into perspective the dual lower bounds output by ICP as described in Section 2.2, we compare this algorithm, firstly, to the cutting plane algorithm for (P_{CYC}) of Kappes et al. (2015), with Gurobi for solving the LPs (denoted here by LP) and, secondly, to the message passing algorithm of Swoboda and Andres (2017), applied to (P_{CYC}) (denoted here by MPC).

Results are shown in Figure 2.2 and Table 2.1. It can be seen from the figure and the table that, for the larger and harder instances *Epinions* and *Slashdot*, ICP converges at under 10^2 seconds, outputting lower bounds that are matched and exceeded by MPC at around 10^3 seconds. It can be seen from Table 2.1 that the situation is similar for the smaller instances: The lower bounds output by ICP are a bit worse than those output by LP or MPC (here compared to the best optimal solution known) but are obtained faster (by as much as three orders of magnitude for *Knott-3D-450*). We conclude that ICP is capable of computing non-trivial lower bounds fast. For the extended version of our heuristic (OW+ICP), it can be seen that the runtime generally increases compared to ICP, in particular on the larger instances. However, the gaps are not improved with the exception of the complete graphs from *Modularity Clustering*, which suggests that the extended packing algorithm is impractical.

In order to study the re-weighting described in Section 2.2.2, we measure its effect on heuristic algorithms for finding feasible solutions. To this end, we employ the implementations of GAEC and of KLj from Levinkov et al. (2017), described in Section 1.2.5.

[1]This results in 2703 edges of cost 0 for *Epinions*, and 1949 such edges for *Slashdot*.

Table 2.2: The table reports the effect of our heuristic re-weighting and a reparameterization with dual solutions obtained from MPC. We report the gaps w.r.t. the original cost function obtained by relation to the objective value of the best known lower bound.

	Image Seg.	*Knott-3D-300*	*Knott-3D-450*	*Mod. Clust.*	*Epinions*	*Slashdot*
GAEC+KLj	0.34%	0.15%	0.07%	1.28%	0.18%	6.69%
ICP+GAEC+KLj	0.10%	0.03%	0.05%	1.14%	0.12%	6.58%
MPC+GAEC+KLj	0.05%	0.01%	0.01%	1.32%	0.09%	6.69%

A comparison between the feasible solutions found by applying the heuristics GAEC and GAEC+KLj to original instances, on the one hand, and to instances re-weighted by ICP with $\alpha = \frac{1}{2}$, on the other hand, can be found in Table 2.2. Note that we only re-weight the input to GAEC and let KLj run with original weights, starting from the solution returned by GAEC, as we found this approach to be advantageous. It can be seen from Table 2.2 that our re-weighting consistently improves the gap. On average, it is slightly less effective than the reparameterization with the more accurate dual solutions obtained from MPC, as proposed by Swoboda and Andres (2017).

2.4 CONCLUSION

We have derived and implemented a combinatorial algorithm to compute heuristic dual solutions for the multicut problem. Empirically, we have shown advantages of our method both in terms of lower bounds and the reduced costs associated with the dual solutions. We conclude that our algorithm can provide valuable dual information fast in practice. For future work, it is relevant to examine if more sophisticated dual solvers such as MPC benefit from a “warm-start” that transforms and exploits the heuristic dual solution.

3 PARTIAL OPTIMALITY

3.1 INTRODUCTION

As computer vision models that employ the multicut or max-cut problem are typically large-scale, standard solution techniques based on solving LP-relaxations do not scale well enough and are thus inapplicable. Even more so, finding globally optimal solutions with branch-and-cut is infeasible with off-the-shelf commercial solvers. Hence, the need arises for developing specialized heuristic solvers that output high-quality solutions for real-world problems, despite the worst-case NP-hardness of the multicut and max-cut problem. Unfortunately, although heuristic solvers often achieve a good empirical performance, they usually come without any optimality guarantees. Specifically, even if large parts of the variable assignments computed by a heuristic agree with globally optimal solutions, such optimality is not recognized.

In this chapter we consider combinatorial techniques for the multicut and max-cut problem by which we can efficiently find *persistency* (a.k.a. partial optimality). Persistent variable assignments come with a certificate that proves their agreement with a globally optimal solution. The potential benefits are twofold: (i) After running a primal heuristic, we can compute certificates which show that some variables are persistent. (ii) Even before running a heuristic, we may determine in a preprocessing step persistent variable assignments. In either case, the problem size can be reduced. In the first case, a subsequent optimization with exact solvers is accelerated. In the second case, possibly also the runtime of a heuristic algorithm is reduced and the solution quality improved.

A joint treatment of the multicut and max-cut problem seems instructive, since many criteria have a similar formulation and are based on analogous arguments. For the max-cut problem we offer, to our knowledge, a novel approach for computing persistent variable assignments. For the multicut problem our empirical evidence suggests that our method offers substantial improvement over prior work on persistency. Our empirical results are most significant for very large scale problems which current heuristics can barely handle, e.g. in biomedical image segmentation (Beier et al., 2017). By reducing problem size via persistency, our method enables high quality solutions in such cases.

The chapter is organized as follows. In Section 3.2 we review the related work. In Section 3.3 we recap the concept of improving mappings in the context of persistency. Further, we introduce fundamental building blocks for the construction of improving mappings for the multicut and max-cut problem. In Section 3.4 and 3.5 we present our combinatorial persistency criteria and devise algorithms to check them. Finally, in Section 3.6 we evaluate our methods in numerical experiments on instances from the literature and compare to related work.

3.2 RELATED WORK

Persistency for Markov Random Fields (MRF) and, as a special case, for the binary quadratic optimization problem (a.k.a. Quadratic Pseudo-Boolean Optimization (QPBO)), has been well studied. It was observed by Nemhauser and Trotter (1975) that a natural LP-relaxation of the stable set problem has the *persistency* property: All integral variables of LP-solutions coincide with a globally optimal one. This result has been transferred to QPBO (Hammer et al., 1984; Boros and Hammer, 2002; Boros et al., 2008) and extended by Wang and Kleinberg (2009) to find relational persistency, i.e. showing that some pairs of variables must have the same/different values. For higher order binary unrestricted optimization problems, the concept of roof duality can be extended to obtain further persistency results (Rother et al., 2007; Kahl and Strandmark, 2012; Kolmogorov, 2012). Going beyond the basic LP-relaxation for QPBO, persistency certificates involving tighter LP-relaxations for higher order polynomial 0/1-programs that do not possess the persistency property (i.e. integral variables need not be persistent) have been studied by Adams et al. (1998).

For general MRFs, criteria that can be elementarily checked include Dead End Elimination (DEE) (Desmet et al., 1992). More powerful techniques generalizing DEE that still can be used for fast preprocessing can be found in (Wang and Zabih, 2016). The MQPBO method (Kohli et al., 2008) consists of transforming multilabel MRFs to the QPBO problem and persistency results from QPBO can subsequently be used to obtain persistency for the original multilabel MRF. Persistency criteria for the multilabel Potts problem that can be efficiently checked with max-flow computations have been developed by Kovtun (2003, 2011) and refined by Gridchyn and Kolmogorov (2013). More powerful criteria based on LP-relaxations have been proposed for the multilabel Potts problem by Swoboda et al. (2013) and by Shekhovtsov (2014); Swoboda et al. (2016); Shekhovtsov et al. (2017) for general discrete MRFs. An in-depth exposition of the concept of improving mappings that is used implicitly or explicitly for all of the above MRF criteria can be found in (Shekhovtsov, 2013). A comprehensive theoretical discussion and comparison of the above persistency techniques can be found in (Shekhovtsov, 2016).

There has been, to our knowledge, less work on persistency for the multicut and max-cut problem. For multicut, the work (Alush and Goldberger, 2012) proposed simple persistency criteria that allow to fix some edge assignments. These criteria are generalized by Lange et al. (2018), which is an earlier version of our work. We are not aware of any persistency results for max-cut. Also it is not easily possible to transfer persistency results from QPBO to max-cut, even though there exist straightforward transformations between these two problems. The underlying reason is that the transformation from max-cut to QPBO introduces symmetries which current persistency criteria cannot handle. More specifically, known persistency criteria rely on an improving mapping, but in symmetric instances it is always possible to map a labeling to an equivalent one with the same cost by exploiting symmetries. Consequently, fixed-points of improving mappings, which amount to persistent variables, cannot be found. For the closely related (yet polynomial-time solvable) min-cut problem, a family of persistency criteria were proposed by Padberg and Rinaldi (1990); Henzinger et al. (2018). They directly translate to the max-cut problem and we derive them as special cases in our study below.

The more involved constraints describing the multicut and max-cut problem make it difficult to directly transfer some of the powerful persistency techniques that are available for MRFs. In our work we show how the framework of improving mappings developed by Shekhovtsov (2013) can be used to derive persistency criteria for combinatorial problems with more complicated constraint structures, such as the multicut and max-cut problem, once a class of mappings that act on feasible solutions is identified. Specifically, we show that the known multicut persistency criteria from (Lange et al., 2018) and the persistency criteria from (Henzinger et al., 2018) (transferred to the max-cut problem) can be derived in our theoretical framework. Moreover, we define more powerful criteria that can find significantly more persistent variables, as shown in the experimental Section 3.6, yet can be evaluated efficiently. We believe that our approach of composing improving mappings from elementary mappings is instructive in the search for more persistency criteria.

3.3 IMPROVING MAPPINGS

In this section, we introduce improving mappings as a concept to derive partial optimality results and define elementary building blocks to construct improving mappings for the multicut and max-cut problem.

As before, let $G = (V, E)$ be a graph and $\theta \in \mathbb{R}^E$. In this chapter we write $X \subseteq \{0,1\}^E$ for a set of feasible characteristic vectors, which may represent either multicuts or cuts, i.e. $X = \mathsf{MC} \cap \mathbb{Z}^E$ or $X = \mathsf{CUT} \cap \mathbb{Z}^E$. For reference throughout this chapter, we introduce the generic problem

$$\min_{x \in X} \sum_{e \in E} \theta_e x_e \tag{P_X}$$

which represents either (P_{MC}) or (P_{CUT}).

Definition 3.1 (Shekhovtsov (2014))**.** A mapping $p\colon X \to X$ with the property

$$\theta^\top p(x) \leq \theta^\top x \quad \forall x \in X$$

is called *improving mapping*.

An improving mapping p that maps some variable x_e to a fixed value β provides partial optimality at x_e: For each feasible element $x \in X$, applying p to x and thus fixing $x_e = \beta$ gives another element that is at least as good. We refer to the assignment $x_e = \beta$, which may be determined by some suboptimal method, as an *persistent* assignment, since it persists at a global optimum.

Lemma 3.1 (Persistency)**.** *Let $p\colon X \to X$ be an improving mapping and $\beta \in \{0,1\}$. If*

$$p(x)_e = \beta \quad \forall x \in X,$$

then $x^*_e = \beta$ in an optimal solution x^* of (P_X).

Proof. Let x be an optimal solution of (P_X). Then $x^* = p(x)$ is also optimal and $x^*_e = \beta$. □

There are two trivial improving mappings: (i) The identity mapping $\mathrm{id}\colon x \mapsto x$. It does not provide any persistency at all, given that no variable is fixed by the constraint $x \in X$ alone. (ii) The mapping $p^*\colon x \mapsto x^*$ that maps any x to a fixed optimal solution $x^* \in \operatorname{argmin}_{x \in X} \theta^\top x$. This mapping obviously provides the maximal persistency, i.e. it fixes all variables, but for NP-hard problems it is generally intractable to compute x^*.

We are hence interested in a middle ground: We want to find improving mappings that fix as many variables as possible (unlike id) but that are computable in polynomial time (unlike p^*). This allows us to simplify the original problem (P_X) by fixing the persistent variables. For the multicut problem we can contract those edges that can be persistently set to 0, which allows to shrink the underlying graph. For the max-cut problem, however, any value for persistent variables can be exploited for contractions, as we note below.

3.3.1 Elementary Mappings

In order to construct improving mappings for the multicut and max-cut problem, we employ the elementary mappings defined in this section.

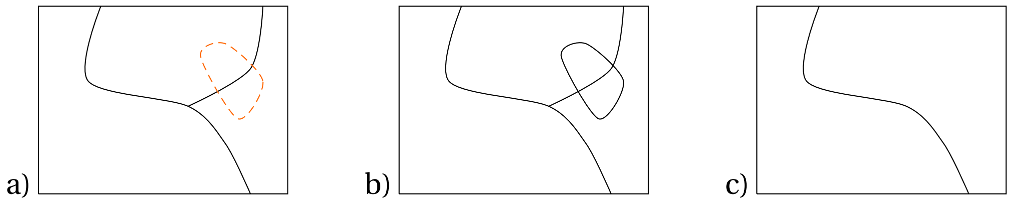

Figure 3.1: Illustration of elementary mappings. a) Original multicut $x \in \mathrm{MC} \cap \mathbb{Z}^E$ (solid lines) and connected region U (dashed line). b) Result of cut mapping $p_{\delta(U)}(x)$. c) Result of join mapping $p_U(x)$.

Definition 3.2 (Multicut mappings)**.** Let $U \subseteq V$ be a set of nodes that induce a connected component of G.

(i) The *elementary cut mapping* $p_{\delta(U)}$ is defined as

$$p_{\delta(U)}(x) = x \vee \mathbb{1}_{\delta(U)}\,.$$

In other words, this means that $p_{\delta(U)}(x)_e = 1$ for all edges $e \in \delta(U)$ and $p_{\delta(U)}(x)_e = x_e$ otherwise.

(ii) The *elementary join mapping* p_U is defined as

$$p_U(x)_{uv} = \begin{cases} 0, & uv \in E(U) \\ 0, & \exists uv\text{-path } P \text{ such that for all } e \in P: \\ & x_e = 0 \text{ or } e \in E(U) \\ x_{uv}, & \text{otherwise.} \end{cases} \tag{3.1}$$

Intuitively, the elementary cut mapping $p_{\delta(U)}$ adds the cut $\delta(U)$ to the multicut defined by x. The elementary join mapping p_U merges all components that intersect with U, cf. Figure 3.1. To show well-definedness of the elementary cut and join mapping rigorously, recall the characterization of multicuts in terms of intersections with cycles from Lemma 1.1.

Lemma 3.2 (Well-definedness)**.** *The mappings $p_{\delta(U)}$ and p_U are well-defined, i.e.*

(i) $p_{\delta(U)}\colon \mathsf{MC}\cap\mathbb{Z}^E \to \mathsf{MC}\cap\mathbb{Z}^E$ *for any connected* $U\subseteq V$

(ii) $p_U\colon \mathsf{MC}\cap\mathbb{Z}^E \to \mathsf{MC}\cap\mathbb{Z}^E$ *for any connected* $U\subseteq V$.

Proof. (i) Let $x\in\mathsf{MC}\cap\mathbb{Z}^E$ and assume that $z=p_{\delta(U)}(x)\notin\mathsf{MC}\cap\mathbb{Z}^E$. Then there exists a cycle C with exactly one cut edge in z, i.e. $z_f=1$ for some $f\in C$ and $z_e=0$ for all $e\in C\setminus\{f\}$. It holds that $x_f=1$ and thus C crosses $\delta(U)$ exactly once, which is impossible.

(ii) Let $x\in\mathsf{MC}\cap\mathbb{Z}^E$ and assume that $z=p_U(x)\notin\mathsf{MC}\cap\mathbb{Z}^E$. Then there is a cycle C with $z_f=1$ for some $f\in C$ and $z_e=0$ for all $e\in C\setminus\{f\}$. Since $z\le x$ there exists an edge $uv=g\in C$, $g\neq f$ with $x_g=1$ and $z_g=0$. Then, according to (3.1), there exists a uv-path P such that $x_e=0$ for all $e\in P\setminus E(U)$. Replace the cycle C with the cycle induced by $C\triangle(P\cup\{g\})$. Repeating this argument for all such edges $g\in C$ yields a path P' connecting the endpoints of f such that $x_e=0$ for all $e\in P'\setminus E(U)$, which is a contradiction to $z_f=1$. □

The elementary mapping for the max-cut problem exploits the well-known property of cuts that they are closed under taking symmetric differences (of edges).

Fact 3.1 (Schrijver (2003))**.** Let $x,y\in\mathsf{CUT}\cap\mathbb{Z}^E$. Then $x\triangle y\in\mathsf{CUT}\cap\mathbb{Z}^E$.

In particular, since $x\mapsto x\triangle y$ is an involution (i.e. its own inverse) for any cut $y\in\mathsf{CUT}\cap\mathbb{Z}^E$, it holds that $\mathsf{CUT}\cap\mathbb{Z}^E\triangle y=\{x\triangle y\mid x\in\mathsf{CUT}\cap\mathbb{Z}^E\}=\mathsf{CUT}\cap\mathbb{Z}^E$. Given an instance of max-cut defined by $G=(V,E),\theta$ and a cut $y\in\mathsf{CUT}\cap\mathbb{Z}^E$, this transformation of the feasible set corresponds to *switching* the signs of θ_e for all $e\in E$ with $y_e=1$ and adding the constant $\sum_{e\in E}\theta_e y_e$ to the objective value. If y is optimal for the original instance, then $y\triangle y=0$ is optimal for the transformed instance. Hence, whenever we want to compute persistency for $x_f=1$, we can transform the instance to an equivalent one by applying the described switching for any cut that contains f and then checking whether $x_f=0$ holds persistently.

a) b)

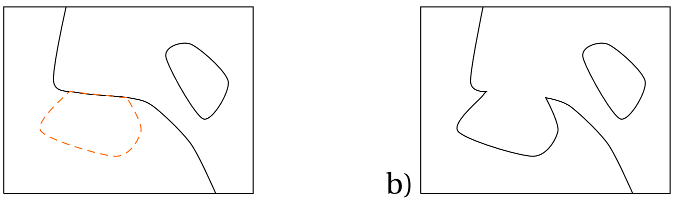

Figure 3.2: Illustration of symmetric difference mapping. a) Original cut $x\in\mathsf{CUT}\cap\mathbb{Z}^E$ (solid lines) and cut $\delta(U)$ (dashed orange line). b) Result of symmetric difference mapping $p^{\triangle}_{\delta(U)}(x)$.

Definition 3.3 (Symmetric Difference Mapping)**.** Let $U \subseteq V$. The *elementary symmetric difference mapping* $p^{\triangle}_{\delta(U)}$ w.r.t. $\delta(U)$ is defined as

$$p^{\triangle}_{\delta(U)}(x) = x \triangle \mathbb{1}_{\delta(U)}\,.$$

In other words, this means that $p^{\triangle}_{\delta(U)}(x)_e = 1 - x_e$ for all edges $e \in \delta(U)$ and $p^{\triangle}_{\delta(U)}(x)_e = x_e$ otherwise. The symmetric difference mapping is well-defined because of Fact 3.1. See Figure 3.2 for an illustration of $p^{\triangle}_{\delta(U)}$.

3.4 PERSISTENCY CRITERIA

In this section, we propose subgraph-based criteria for finding improving mappings. We provide criteria for small connected subgraphs such as edges or triangles as well as criteria for general connected subgraphs. In Section 3.5, we present efficient algorithms to check the subgraph criteria proposed in this section.

A basic persistency criterion for some variable x_e requires that x_e is not constrained by the condition $x \in X$. Then, for any optimal solution x^* of (P_X) it holds that $x^*_e = 0$ if $\theta_e > 0$ and $x^*_e = 1$ if $\theta_e < 0$. In the multicut problem this is the case if the edge e is not contained in any conflicted cycle, i.e. any cycle with exactly one negative edge (cf. Section 1.2.4). There are (at least) two ways this can happen: 1. An edge $e \in E$ is not contained in any cycle at all, in other words e is a *bridge*. 2. The endpoints of a negative edge $e \in E^-$ belong to distinct components of $G^+ = (V, E^+)$. Thus, we can restrict the instance of the problem to the maximal components G that are connected in G^+ and biconnected in G. These basic conditions were observed by Alush and Goldberger (2012).

3.4.1 Edge Criterion

Our first instructive criterion (Theorem 3.1) requires that the cost associated with a particular edge $f \in E$ outweighs the total cost of other edges in a cut that contains f. If an edge f satisfies the condition specified by Theorem 3.1 we call it *dominant*. The persistency criterion generalizes the basic conditions stated above, since each edge that is not contained in any conflicted cycle is also dominant. See Figure 3.3 for an example of a dominant edge.

Theorem 3.1 (Edge Criterion)**.** *Let $f \in E$ be an edge and $U \subseteq V$ be connected with $f \in \delta(U)$. Further, let $\beta = (1 - \operatorname{sign}\theta_f)/2$. If*

$$\begin{cases} \theta_f \geq \sum_{e \in \delta(U)\setminus\{f\}} |\theta_e|, & \mathrm{P}_X = \mathrm{P}_{\mathsf{MC}},\ \beta = 0 \qquad (3.2)\\ |\theta_f| \geq \sum_{e \in \delta(U)\cap E^+} \theta_e, & \mathrm{P}_X = \mathrm{P}_{\mathsf{MC}},\ \beta = 1 \qquad (3.3)\\ |\theta_f| \geq \sum_{e \in \delta(U)\setminus\{f\}} |\theta_e|, & \mathrm{P}_X = \mathrm{P}_{\mathsf{CUT}} \qquad (3.4) \end{cases}$$

then $x^*_f = \beta$ in some optimal solution x^* of (P_X).

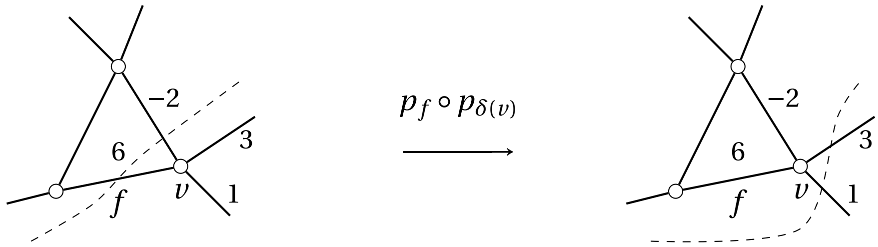

Figure 3.3: In this example, the edge f is positive dominant w.r.t. the cut $\delta(v)$ according to Theorem 3.1. The mapping $p_f \circ p_{\delta(v)}$ maps some multicut on the left (indicated by the dashed line), which cuts f, to another multicut on the right that has the same objective value but does not cut f.

Proof. First, we show that the mapping

$$p(x) = \begin{cases} p^{\triangle}_{\delta(U)}(x) & \text{if } x_f \neq \beta \\ x & \text{else} \end{cases}$$

is improving for the max-cut problem. Let $x \in \mathrm{CUT} \cap \mathbb{Z}^E$, $z = p(x)$ and suppose $x_f \neq \beta$. It holds that

$$\begin{aligned} \theta^\top z - \theta^\top x &= \theta_f(z_f - x_f) + \sum_{e \in \delta(U) \setminus \{f\}} \theta_e(z_e - x_e) \\ &= -|\theta_f| + \sum_{e \in \delta(U) \setminus \{f\}} \theta_e(z_e - x_e) \\ &\leq -|\theta_f| + \sum_{e \in \delta(U) \setminus \{f\}} |\theta_e| \\ &\leq 0. \end{aligned}$$

Similarly, for $\beta = 0$, we show that the mapping

$$p(x) = \begin{cases} (p_f \circ p_{\delta(U)})(x) & \text{if } x_f \neq \beta \\ x & \text{else} \end{cases}$$

is improving for the multicut problem. Let $x \in \mathrm{MC} \cap \mathbb{Z}^E$, $z = p(x)$ and suppose $x_f \neq \beta$. It holds that

$$\begin{aligned} \theta^\top z - \theta^\top x &= \theta_f(0 - 1) + \sum_{e \in \delta(U) \setminus \{f\}} \theta_e(z_e - x_e) \\ &\leq -|\theta_f| + \sum_{e \in \delta(U) \setminus \{f\}} |\theta_e| \\ &\leq 0. \end{aligned}$$

Finally, for $\beta = 1$, we show that the mapping

$$p(x) = \begin{cases} p_{\delta(U)}(x) & \text{if } x_f \neq \beta \\ x & \text{else} \end{cases}$$

is improving for multicut. Let $x \in \mathrm{MC} \cap \mathbb{Z}^E$, $z = p(x)$ and suppose $x_f \neq \beta$. It holds that

$$\begin{aligned}\theta^\top z - \theta^\top x &= \theta_f(1-0) + \sum_{e \in \delta(U)\setminus\{f\}} \theta_e(1-x_e) \\ &\leq -|\theta_f| + \sum_{e \in \delta(U)\cap E^+} \theta_e(1-x_e) \\ &\leq -|\theta_f| + \sum_{e \in \delta(U)\cap E^+} |\theta_e| \\ &\leq 0.\end{aligned}$$

This concludes the proof. □

Simple candidates for U in Theorem 3.1 are $\{u\}$ and $\{v\}$ where $f = uv$. Checking these for every edge $f \in E$ can be done in linear time in a preprocessing algorithm. Moreover, this algorithm can be implemented to solve (P_X) exactly if G is series-parallel, as we show in Section 3.5. All u-v-cuts (the cuts that separate u from v) can be checked at once by minimizing the right-hand sides of (3.2) – (3.4) via max-flow techniques on G weighted with $|\theta|$, respectively $G^+ = (V, E^+)$ weighted with θ for (3.3). Note that the condition in (3.3) is less restrictive than (3.2). Computing a *cut tree* (Gomory and Hu, 1961) of G w.r.t. $|\theta|$ or G^+ w.r.t. θ reduces the total computational effort of checking the criterion for all edges $f \in E$ to $|V|-1$ max-flow problems.

3.4.2 Subgraph Criteria

We give a technical lemma that allows to generalize the persistency criterion stated in Theorem 3.1.

Lemma 3.3. *Let $f \in E$ and $\beta \in \{0,1\}$. Further, let $H = (V_H, E_H)$ be a connected subgraph of G such that $f \in E_H$. If for every $y \in X(H)$ with $y_f = 1-\beta$, there exists a mapping $p^y \colon X \to X$ such that for all $x \in X$ whose restriction to H agrees with y, i.e. $x_{E_H} = y$, we have*

(i) $\theta^\top p^y(x) \leq \theta^\top x$

(ii) $p^y(x)_f = \beta$,

then $x^*_f = \beta$ in some optimal solution x^*.

Proof. Condition (i) implies that the mapping $p\colon X \to X$ defined by

$$p(x) = \begin{cases} p^y(x) & \text{if } x_{E_H} = y \\ x & \text{else} \end{cases}$$

is improving. Condition (ii) implies $p(x)_e = \beta$ for all x. □

Consider the following special case when H is a triangle subgraph. If the persistency criterion is satisfied, then for every assignment of x_{E_H} there is an improving combination of elementary mappings from Section 3.3.1.

Corollary 3.1 (Triangle Criterion). *Let $\{uw, uv, vw\} \subset E$ be a triangle. Let $U \subset V$ be such that $uv, uw \in \delta(U)$, and $W \subset V$ be such that $uw, vw \in \delta(W)$.*

(i) If

$$\theta_{uw} + \theta_{uv} \geq \sum_{e \in \delta(U) \setminus \{uw, uv\}} |\theta_e|, \tag{3.5}$$

$$\theta_{uw} + \theta_{vw} \geq \sum_{e \in \delta(W) \setminus \{uw, vw\}} |\theta_e| \tag{3.6}$$

holds, then $x^*_{uw} = 0$ for some optimal solution of (P_{CUT}).

(ii) If additionally

$$\theta_{uw} + \theta_{uv} + \theta_{vw} \geq \sum_{e \in \delta(\{u,v,w\}) \cap E^+} \theta_e \tag{3.7}$$

holds, then $x^*_{uw} = 0$ for some optimal solution of (P_{MC}).

Proof. We use Lemma 3.3:

(i) In the case $x_{uw} = 1$, $x_{uv} = 1$, $x_{vw} = 0$ apply $p^{\triangle}_{\delta(U)}$. In the case $x_{uw} = 1$, $x_{uv} = 0$, $x_{vw} = 1$ apply $p^{\triangle}_{\delta(W)}$. These mappings are improving due to (3.5) and (3.6).

(ii) In the case $x_{uw} = 1$, $x_{uv} = 1$, $x_{vw} = 0$ apply $p_{\{u,w\}} \circ p_{\delta(U)}$. In the case $x_{uw} = 1$, $x_{uv} = 0$, $x_{vw} = 1$ apply $p_{\{u,w\}} \circ p_{\delta(W)}$. These mappings are improving analogously to (i). In the additional case $x_{uw} = 1$, $x_{uv} = 1$, $x_{vw} = 1$ apply the mapping $p = p_{\{u,v,w\}} \circ p_{\delta(\{u,v,w\})}$. It is improving, since

$$\begin{aligned}
\theta^\top p(x) - \theta^\top x &= \sum_{e \in \delta(\{u,v,w\})} \theta_e(1 - x_e) - \theta_{uv} - \theta_{uw} - \theta_{vw} \\
&\leq \sum_{e \in \delta(\{u,v,w\}) \cap E^+} \theta_e(1 - x_e) - \sum_{e \in \delta(\{u,v,w\}) \cap E^+} \theta_e \\
&\leq 0.
\end{aligned}$$

□

A straightforward choice for the cuts in Corollary 3.1 are $\delta(\{u\}), \delta(\{w\}), \delta(\{v,w\})$ and $\delta(\{u,v\})$, as depicted in Figure 3.4 a). It is possible to find better cuts w.r.t. costs $|\theta|$, but we are not aware of any more efficient technique than to explicitly compute them via max-flow for every triangle (unlike computing a Gomory-Hu tree to evaluate the single edge criterion for all edges).

We further employ Lemma 3.3 to state general subgraph criteria for the multicut and max-cut problem. They are proved by showing that the mapping $p \colon X \to X$, which suitably applies $p_{V_H} \circ p_{\delta(V_H)}$, respectively $p^{\triangle}_{\delta(U)}$, is improving. See Figure 3.4 b) for a schematic illustration.

Theorem 3.2 (Multicut Subgraph Criterion). *Let $H = (V_H, E_H)$ be a connected subgraph of G and suppose $uv \in E_H$. If*

$$\min_{y \in MC(H)} \theta^\top y = 0 \tag{3.8}$$

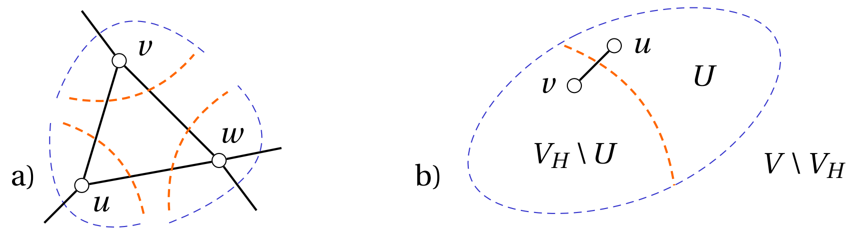

Figure 3.4: a) The conditions presented in Corollary 3.1 compare the costs of inner cuts (- -) and outer cuts (- -) around the triangle $\{u, v, w\}$. b) The conditions (3.9) and (3.11), presented in Theorem 3.2 and 3.3, compare the costs of the inner cut $\delta(U, V_H \setminus U)$ and the outer cut $\delta(V_H) = \delta(U, V \setminus V_H) \cup \delta(V_H \setminus U, V \setminus V_H)$.

and for all $U \subset V_H$ with $u \in U$ and $v \notin U$ it holds that

$$\sum_{e \in \delta(U, V_H \setminus U)} \theta_e \geq \sum_{e \in \delta(V_H) \cap E^+} \theta_e, \tag{3.9}$$

then $x^*_{uv} = 0$ in some optimal solution x^* of ($\mathrm{P_{MC}}$).

Proof. We use Lemma 3.3. Let $y \in \mathrm{MC}(H) \cap \mathbb{Z}^{E_H}$ with $y_{uv} = 1$ and suppose $x \in \mathrm{MC} \cap \mathbb{Z}^E$ with $x_{E_H} = y$. Then there is a multicut M of H such that $y = \mathbb{1}_M$. Due to (3.8), every (multi-)cut of H has nonnegative weight. Therefore, there exists some $U \subset V_H$ with $u \in U$ and $v \notin U$ such that $\delta(U, V_H \setminus U) \subseteq M$ and

$$\sum_{e \in E_H} \theta_e x_e = \sum_{e \in M} \theta_e \geq \sum_{e \in \delta(U, V_H \setminus U)} \theta_e. \tag{3.10}$$

Let $p^y(x) = (p_{V_H} \circ p_{\delta(V_H)})(x)$, then it follows from (3.9) and (3.10) that

$$\begin{aligned} \theta^\top p^y(x) - \theta^\top x &= \sum_{e \in \delta(V_H)} \theta_e (1 - x_e) - \sum_{e \in E_H} \theta_e x_e \\ &\leq \sum_{e \in \delta(V_H) \cap E^+} \theta_e (1 - x_e) - \sum_{e \in \delta(V_H) \cap E^+} \theta_e \\ &\leq 0. \end{aligned}$$

□

Note that the multicut subgraph criterion stated in Theorem 3.2 is different from the edge and triangle criteria when evaluated on these special subgraphs. If $H = (f, \{f\})$ for some edge $f \in E$, then condition (3.9) translates to

$$\theta_f \geq \sum_{e \in \delta(f) \cap E^+} \theta_e.$$

If H is a triangle, i.e. $H = (\{u, v, w\}, \{uv, uw, vw\})$ for some vertices $u, v, w \in V$, then condition (3.9) translates to

$$\min\{\theta_{uv} + \theta_{uw}, \theta_{uv} + \theta_{vw}, \theta_{uw} + \theta_{vw}\} \geq \sum_{e \in \delta(\{u,v,w\}) \cap E^+} \theta_e.$$

Theorem 3.3 (Max-Cut Subgraph Criterion)**.** *Let $H = (V_H, E_H)$ be a connected subgraph of G and suppose $uv \in E_H$. If for all $U \subset V_H$ with $u \in U$ and $v \notin U$ it holds that*

$$\sum_{e \in \delta(U, V_H \setminus U)} \theta_e \geq \min \left\{ \sum_{e \in \delta(U, V \setminus V_H)} |\theta_e|, \sum_{e \in \delta(V_H \setminus U, V \setminus V_H)} |\theta_e| \right\}, \tag{3.11}$$

then $x^*_{uv} = 0$ in some optimal solution x^* of ($\mathrm{P_{CUT}}$).

Proof. We use Lemma 3.3. Suppose $y \in \mathrm{CUT}(H) \cap \mathbb{Z}^{E_H}$ with $y_{uv} = 1$. Let $U \subset V_H$ be such that y is the incidence vector of $\delta(U, V_H \setminus U)$ in H and suppose $x \in \mathrm{CUT} \cap \mathbb{Z}^E$ with $x_{E_H} = y$. We may assume that

$$\sum_{e \in \delta(U, V_H \setminus U)} \theta_e \geq \sum_{e \in \delta(U, V \setminus V_H)} |\theta_e|,$$

otherwise redefine $U := V_H \setminus U$. Now, let $z = p^y(x) = p^{\triangle}_{\delta(U)}(x)$, then it follows that

$$\begin{aligned} \theta^\top z - \theta^\top x &= \sum_{e \in \delta(U, V_H \setminus U)} \theta_e (0 - 1) + \sum_{e \in \delta(U, V \setminus V_H)} \theta_e (z_e - x_e) \\ &\leq \sum_{e \in \delta(U, V_H \setminus U)} -\theta_e + \sum_{e \in \delta(U, V \setminus V_H)} |\theta_e| \\ &\leq 0. \end{aligned}$$

□

Note that if H is a single edge or a triangle, the subgraph criterion stated in Theorem 3.3 specializes to the edge criterion, respectively triangle criterion, where only the cuts $\delta(\{u\})$, $\delta(\{v\})$, respectively $\delta(\{u\}), \delta(\{w\}), \delta(\{v, w\})$ and $\delta(\{u, v\})$ are considered.

3.4.3 Improved Multicut Subgraph Criterion

In this section we describe a technical improvement of the multicut subgraph criterion presented in Theorem 3.2. Here, improvement means relaxing the inequality (3.9) such that it applies more often (without compromising the persistency result).

To this end, we need to introduce some more notation. For any set of vertices $U \subseteq V$, let

$$\partial U = \{v \in V \mid \exists uv \in E \text{ with } u \in U\}$$

denote the *boundary* of U in V. The boundary of U consists of those vertices in V that have a neighbor in U but are not in U themselves. For any set $U \subseteq V$, its *closure* $\overline{U}$ is defined as the union of U with its boundary, i.e.

$$\overline{U} = U \cup \partial U.$$

Further, for any connected subgraph $H = (V_H, E_H)$, we define its *positive closure* as the subgraph

$$\overline{H} = (\overline{V_H}, E_H \cup (\delta(V_H) \cap E^+)),$$

which additionally includes all positive edges between V_H and its boundary.

Below we state a more refined version of Theorem 3.2. The difference in Theorem 3.4 is that the inner cut is w.r.t. the subgraph $\overline{H}$ instead of H.

Theorem 3.4 (Multicut Subgraph Criterion)**.** *Let $H = (V_H, E_H)$ be a connected subgraph of G and suppose $uv \in E_H$. If*

$$\min_{y \in \mathrm{MC}(H)} \langle \theta, y \rangle = 0$$

and for all $U \subset \overline{V_H}$ with $u \in U$ and $v \notin U$ it holds that

$$\sum_{e \in \delta(U, \overline{V_H} \setminus U) \cap E_{\overline{H}}} \theta_e \geq \sum_{e \in \delta(V_H) \cap E^+} \theta_e, \tag{3.12}$$

then $x^*_{uv} = 0$ in some optimal solution x^* of (P_{MC}).

Proof. The proof is largely analogous to the proof of Theorem 3.2. Suppose $x \in \mathrm{MC} \cap \mathbb{Z}^E$ with $x_{E_H} = y \in \mathrm{MC}(H) \cap \mathbb{Z}^{E_H}$ and $y_{uv} = 1$. Apparently, there exists a multicut M of $\overline{H}$ which extends y such that $x_{E_{\overline{H}}} = \mathbb{1}_M$. Similarly to before, there exists some $U \subset \overline{V_H}$ with $u \in U$ and $v \notin U$ such that $\delta(U, \overline{V_H} \setminus U) \cap E_{\overline{H}} \subseteq M$ and

$$\sum_{e \in \delta(V_H) \cap E^+} \theta_e x_e + \sum_{e \in E_H} \theta_e x_e \geq \sum_{e \in \delta(U, \overline{V_H} \setminus U) \cap E_{\overline{H}}} \theta_e. \tag{3.13}$$

Eventually, using (3.12) and (3.13), we show that the mapping $p^y = (p_{V_H} \circ p_{\delta(V_H)})$ still improves x, as follows:

$$\begin{aligned}
\theta^\top p^y(x) - \theta^\top x &= \sum_{e \in \delta(V_H) \cap E^+} \theta_e (1 - x_e) + \sum_{e \in \delta(V_H) \cap E^-} \theta_e (1 - x_e) - \sum_{e \in E_H} \theta_e x_e \\
&\leq \sum_{e \in \delta(V_H) \cap E^+} \theta_e - \sum_{e \in \delta(V_H) \cap E^+} \theta_e x_e - \sum_{e \in E_H} \theta_e x_e \\
&\leq \sum_{e \in \delta(V_H) \cap E^+} \theta_e - \sum_{e \in \delta(V_H) \cap E^+} \theta_e \\
&= 0.
\end{aligned}$$

□

The inequality (3.12) is less restrictive than (3.9) in Theorem 3.2, because the left-hand side is potentially larger. Indeed, if two neighboring nodes $u, v \in V_H$ are connected by positive edges to some vertex $w \in \partial V_H$ in the boundary (i.e. they form a triangle), then the extension of any cut that separates u from v has to cut another edge of the triangle. Thus, the cost of this edge can be subtracted from the right-hand side of the inequality (3.9) or, equivalently, added to the left-hand side, which is what (3.12) achieves.

For the special case of a single edge subgraph $H = (\{u, v\}, \{uv\})$ the refined condition is explicitly stated as

$$\theta_{uv} \geq \sum_{e \in \delta(uv) \cap E^+} \theta_e - \sum_{w \neq u, v \mid uw, vw \in E^+} \min\{\theta_{uw}, \theta_{vw}\}.$$

Note that the refined condition (3.12) can be checked with the same algorithms presented in Section 3.5 for the condition (3.9) by suitably replacing H with $\overline{H}$. For the sake of clarity, we omit this technical detail from the presentation in the remainder of this chapter.

3.5 ALGORITHMS

In this section we devise algorithms that verify, for a given instance of the multicut or max-cut problem, the persistency criteria presented in Section 3.4. Our method applies all subroutines repeatedly until no more persistent edges are found. An earlier version of our method (Lange et al., 2018) only applies the linear time techniques derived from our criteria.

The subgraph criteria restricted to edges or triangles of G can be checked explicitly by enumeration. Note that listing all triangles of a graph can be done efficiently (Schank and Wagner, 2005).

In the following we outline algorithms for important special cases of the edge criterion as well as a method to check the general subgraph criteria introduced in Section 3.4.

3.5.1 Sparse Cuts

In practice, it is expected that dominant edges are more likely to be found in cuts that are relatively sparse. We discuss two special cases of sparse cuts that are of particular interest, due to the following reasons. First, they can be checked in linear time, which gives rise to a fast preprocessing algorithm. Second, we show that our techniques solve (P_X) to optimality if G is series-parallel.

Two-Edge Cuts Suppose $\delta(U) = \{e, f\}$ is a two-edge cut of G. Apparently, according to Theorem 3.1, at least one of them must be dominant. Further, it is guaranteed that we can simplify the instance by edge deletions or contractions. For the max-cut problem this is clear due to the switching operation noted in Section 3.3.1. For the multicut problem the switching operation is in general invalid. Since $\delta(U)$ only contains two elements, however, switching signs along $\delta(U)$ does not change the set of conflicted cycles and is thus valid. More precisely, we distinguish the following cases for multicut. If both e and f are negative, then both of them are dominant and we can delete them, as they are not contained in any conflicted cycle. If f is positive dominant, we can contract f. If f is negative dominant and e is positive, then we can switch the signs of their coefficients and redefine $x_f := 1 - x_f$ as well as $x_e := 1 - x_e$. Afterwards, the edge f is positive dominant and we can contract f. The two-edge cuts of G can be found in linear time, by computing the 3-edge-connected components of G, cf. (Mehlhorn et al., 2017).

Single-Node Cuts For any $v \in V$ it is easily decided, whether the single-node cut $\delta(v) = \delta(\{v\})$ contains a dominant edge, by considering all edges incident to v. Moreover, if $\deg v = 2$, then $\delta(v)$ is also a two-edge cut and we can apply the switching operation described in the last paragraph even for the multicut problem. Updating the graph and applying these techniques recursively as specified in Algorithm 2 takes linear time. Note that Algorithm 2 is specified for (P_{MC}). It is straightforward to adapt it to (P_{CUT}) by applying switching operations whenever a dominant edge is found such that the edge can be contracted. We have the following theoretical consequence.

Algorithm 2 Single-Node Cut Preprocessing for Multicut

input $G = (V, E), \theta \colon E \to \mathbb{R}$
Initialize objective value offset $\Delta = 0$.
Initialize a queue $Q = V$.
while $Q \neq \emptyset$ **do**
  Extract a vertex $v \in Q$.
  if $\deg v = 1$ **then**
    Get neighbor $u \in V$.
    if $\theta_{uv} \geq 0$ **then**
      Set $x_{uv} = 0$ and contract $uv \in E$.
    else
      Set $x_{uv} = 1$, $\Delta = \Delta + \theta_{uv}$ and delete $uv \in E$.
    end if
  else if $\deg v = 2$ **then**
    Get neighbors $u, w \in V$ with $|\theta_{uv}| \geq |\theta_{wv}|$.
    if $uv \in E^+$ **then**
      Set $x_{uv} = 0$ and contract $uv \in E$.
    else if $uv \in E^-$ and $wv \in E^-$ **then**
      Adjust offset $\Delta = \Delta + \theta_{uv} + \theta_{wv}$.
      Set $x_{uv} = x_{wv} = 1$ and delete $uv, wv \in E$.
    else if $uv \in E^-$ and $wv \in E^+$ **then**
      Adjust offset $\Delta = \Delta + \theta_{uv} + \theta_{wv}$.
      Redefine $x_{uv} = 1 - x_{uv}$, $x_{wv} = 1 - x_{wv}$ and $\theta_{uv} = -\theta_{uv}$, $\theta_{wv} = -\theta_{wv}$.
      Set $x_{uv} = 0$ and contract $uv \in E$.
    end if
  else if $\exists f \in \delta(v)$ positive dominant **then**
    Set $x_f = 0$ and contract $f \in E$.
  end if
  Add to Q all vertices $u \notin Q$ whose neighborhood was changed.
end while
return G, θ, x, Δ

Corollary 3.2. *If G has treewidth at most 2, then Algorithm 2 can be implemented to solve (P_X) exactly in $\mathcal{O}(|V|)$ time.*

Proof. Place the vertices of G into buckets of ascending degree and always pick a vertex of minimal degree. Every graph of treewidth 2 has a vertex v with $\deg v \leq 2$. Since Algorithm 2 only contracts or deletes edges, fixing the variables according to Theorem 3.1, the updated graph still has treewidth at most 2. The number of nodes decreases by 1 in every iteration, hence the algorithm terminates in $\mathcal{O}(|E|) = \mathcal{O}(|V|)$ steps and outputs an optimal solution. □

3.5.2 General Subgraph Criteria

In this section we focus on developing efficient algorithms that find subgraphs H which qualify for the subgraph criteria from Theorem 3.2 and 3.3. Specifically, we propose routines that (i) check for a given connected subgraph H whether some persistency criteria apply and (ii) find good candidates for H.

Subgraph Evaluation Let $H = (V_H, E_H)$ be a subgraph of G that we want to check for persistency condition (3.9), respectively (3.11). Now, for a given edge $uv \in E_H$, we can determine if (3.9) holds true for all $U \subset V_H$ with $u \in U$ and $v \notin U$ by minimizing the left-hand side w.r.t. U. In contrast, for (3.11), we also need to simultaneously maximize the right-hand side, since it depends on U as well. Obviously, minimizing the left-hand side (of either (3.9) or (3.11)) means finding a minimum u-v-cut w.r.t. θ. Further, since the right-hand sides are nonnegative, the minimum u-v-cut must have nonnegative cost. However, in general the costs θ on H may be negative, which renders both optimization problems hard in general.

For this reason, we simplify the problem by restriction to suitable subgraphs H that satisfy Assumption 3.1 below. We shall see subsequently how to utilize this condition. In order to state Assumption 3.1 rigorously, recall from Section 2.2 the packing dual of the multicut problem

$$\begin{aligned} \max \quad & \mathbb{1}^\top \lambda && && (3.14) \\ \text{s.t.} \quad & \sum_{C: e \in C} \lambda_C \le |\theta_e| && \forall e \in E \\ & \lambda \ge 0 \end{aligned}$$

and the reduced costs for ($\mathrm{P_{MC}}$) associated with any dual feasible $\lambda \ge 0$, defined by

$$\tilde{\theta}_e = \Big(|\theta_e| - \sum_{C: e \in C} \lambda_C \Big) \operatorname{sign} \theta_e \qquad \forall e \in E. \tag{3.15}$$

Assumption 3.1. Let $H = (V_H, E_H, \theta)$ be a weighted graph such that

i) The graph H has a trivial optimal multicut solution $y^* = 0$:

$$\min_{y \in \mathrm{MC}(H)} \theta^\top y = 0 = \theta^\top y^*.$$

ii) An optimal packing dual solution λ^* (cf. Section 2.2) that corresponds to y^* for the multicut problem on H is at hand.

Note that Assumption 3.1 i) implies a trivial max-cut solution as well, since $\mathrm{CUT} \subseteq \mathrm{MC}$. Assumption 3.1 has the following expedient consequence.

Lemma 3.4. *Let $H = (V_H, E_H, \theta)$ be a weighted graph that satisfies Assumption 3.1. Then the reduced costs $\tilde{\theta}$ associated with λ^*, defined by (3.15), satisfy $\tilde{\theta}_e \ge 0$ for all $e \in E_H$ and for any cut $\delta(U)$ of H it holds that*

$$0 \le \sum_{e \in \delta(U)} \tilde{\theta}_e \le \sum_{e \in \delta(U)} \theta_e. \tag{3.16}$$

Proof. As H satisfies Assumption 3.1, the dual problem (3.14) evaluates to zero. Thus, since any conflicted cycle contains exactly one edge $e \in E_H$ with $\theta_e < 0$, we must have $\sum_{C:e\in C} \lambda^*_C = |\theta_e|$, which implies $\tilde{\theta}_e = 0$. Furthermore, for any cut $\delta(U)$ of H it holds that

$$\begin{aligned}
\sum_{e\in\delta(U)} \tilde{\theta}_e &= \sum_{e\in\delta(U)\cap E^+} \tilde{\theta}_e + \sum_{e\in\delta(U)\cap E^-} \tilde{\theta}_e \\
&= \sum_{e\in\delta(U)\cap E^+} \Big(\theta_e - \sum_{C:e\in C} \lambda^*_C\Big) + \sum_{e\in\delta(U)\cap E^-} \Big(\theta_e + \sum_{C:e\in C} \lambda^*_C\Big) \\
&= \sum_{e\in\delta(U)} \theta_e + \sum_{\substack{e\in\delta(U)\cap E^- \\ C:e\in C}} \lambda^*_C - \sum_{\substack{e\in\delta(U)\cap E^+ \\ C:e\in C}} \lambda^*_C \\
&\le \sum_{e\in\delta(U)} \theta_e .
\end{aligned}$$

The last inequality holds true, because every cycle with precisely one negative edge e, where $e \in \delta(U) \cap E^-$, also contains some positive edge $f \in \delta(U) \cap E^+$, as it crosses $\delta(U)$ at least twice. This concludes the proof. □

Our method exploits Assumption 3.1 and Lemma 3.4 as follows. First, we compute a heuristic solution to the packing dual (3.14) with the ICP algorithm from Chapter 2. Then, if the computed dual bound shows that H has a trivial multicut solution (and thus the dual solution is optimal), we can compute lower bounds to the left-hand side of (3.9) and (3.11) by applying max-flow techniques on H with capacities $\tilde{\theta}$.

In the case of the multicut problem, the right-hand side of (3.9) is constant w.r.t. U so it suffices to compute a Gomory-Hu tree on H. In the case of the max-cut problem, however, this is not sufficient, since the right-hand side of (3.11) also depends on U. Here, after replacing θ_e by $\tilde{\theta}_e$ for all $e \in E_H$, we need to solve the following min max problem

$$\begin{aligned}
&\min_{\substack{U\subset V_H: \\ u\in U, v\notin U}} \Bigg(\sum_{e\in\delta(U,V_H\setminus U)} \tilde{\theta}_e - \min\Bigg\{ \sum_{e\in\delta(U,V\setminus V_H)} |\theta_e|, \sum_{e\in\delta(V_H\setminus U,V\setminus V_H)} |\theta_e| \Bigg\}\Bigg) \\
&= -\sum_{e\in\delta(V_H)} |\theta_e| + \min_{\substack{U\subset V_H: \\ u\in U, v\notin U}} \Bigg(\sum_{e\in\delta(U,V_H\setminus U)} \tilde{\theta}_e + \max\Bigg\{ \sum_{e\in\delta(U,V\setminus V_H)} |\theta_e|, \sum_{e\in\delta(V_H\setminus U,V\setminus V_H)} |\theta_e| \Bigg\}\Bigg). \quad (3.17)
\end{aligned}$$

As solving this problem exactly appears to be difficult, we propose to solve a relaxation that is obtained by replacing the inner max term with

$$\max_{\alpha\in[0,1]} \alpha \sum_{e\in\delta(U,V\setminus V_H)} |\theta_e| + (1-\alpha) \sum_{e\in\delta(V_H\setminus U,V\setminus V_H)} |\theta_e|$$

and then swapping the order of min and max. This yields a lower bound to (3.17) of

$$-\sum_{e\in\delta(V_H)} |\theta_e| + \max_{\alpha\in[0,1]} \min_{\substack{U\subset V_H: \\ u\in U, v\notin U}} \Bigg(\sum_{e\in\delta(U,V_H\setminus U)} \tilde{\theta}_e + \alpha \sum_{e\in\delta(U,V\setminus V_H)} |\theta_e| + (1-\alpha) \sum_{e\in\delta(V_H\setminus U,V\setminus V_H)} |\theta_e| \Bigg). \quad (3.18)$$

The problem (3.18) is the maximization of a concave, non-smooth function on the unit interval, which can be solved efficiently with the bisection method. In every iteration, the inner minimization problem needs to be solved for a fixed $\alpha \in [0,1]$, which can be formulated again as a max-flow problem.

For solving the max-flow problems that occur in our method, we use an implemenation of Boykov-Kolmogorov's algorithm with reused search trees (Boykov and Kolmogorov, 2004; Kohli and Torr, 2005). For computing Gomory-Hu trees, we use a parallelized implementation of Gusfield's algorithm (Gusfield, 1990; Cohen et al., 2011).

Finding Candidate Subgraphs To efficiently find good candidate subgraphs, we employ the following strategy. First, we compute a primal feasible solution $\bar{x} \in X$ by a fast heuristic method such as greedy edge contraction algorithms (Kahruman-Anderoglu et al., 2007; Keuper et al., 2015b), cf. Section 1.2.5. If the heuristic solution $\bar{x}$ is reasonably good, then many components defined by $\bar{x}$ should be close to optimal. Thus, in the case of the multicut problem, the components may already serve as candidate subgraphs. In the case of the max-cut problem, we use $\bar{x}$ to transform the instance by the switching operation described in Section 3.3.

Then, we compute a heuristic packing dual solution $\bar{\lambda}$ with ICP for the entire graph G. The candidate subgraphs are determined as the connected components of the positive residual graph $(V, \{e \in E \mid \tilde{\theta}_e > 0\})$, where $\tilde{\theta}$ is defined as before in (3.15). The intuition behind this strategy is that, by construction, the edges within the subgraphs have relatively higher weight than the outgoing edges. This facilitates the application of the conditions (3.9) and (3.11).

Reduced Cost Fixing Further, whenever both a primal solution and dual solution are available, we use the following technique known as *reduced cost fixing* (Balas and Martin, 1980) to determine additional persistent variables. Let $\gamma = \theta^\top x - \mathbb{1}^\top \bar{\lambda} - L^{\text{triv}}$ denote the duality gap of the primal-dual solution pair and suppose $\gamma < \tilde{\theta}_f$ for some $f \in E$. Then, it follows that $x_f = 1$ cannot be optimal and thus we can fix $x_f = 0$.

3.6 EXPERIMENTS

In order to study the effectiveness of our methods, we evaluate them on a collection of more than 200 instances from the literature. The size of the instances ranges from a few hundred to hundreds of millions of variables (edges). We measure and compare the average relative size reduction of test instances that is obtained by applying our algorithms.

Instances For the multicut problem we use segmentation and clustering instances from the OpenGM benchmark (Kappes et al., 2015) as well as biomedical segmentation instances provided by the authors of (Beier et al., 2017) and (Pape et al., 2017). The dataset *Image Segmentation* contains planar graphs that are constructed from superpixel adjacencies of photographs. The *Knott-3D* data sets contains non-planar graph arising from volume images acquired by electron microscopy. The set *Modularity Clustering* contains complete graphs constructed from clustering problems on small social networks. The *CREMI* data sets contain supervoxel adjacency graphs obtained from volume image scans of neural

Table 3.1: The table gathers for each data set the number of instances (#I), the graph sizes and instance type (P_X).

Data set	#I	$\lvert V\rvert$	$\lvert E\rvert$	P_X
Image Segmentation	100	156–3764	439–10970	P_{MC}
Knott-3D-150	8	572–972	3381–5656	P_{MC}
Knott-3D-300	8	3846–5896	23k–36k	P_{MC}
Knott-3D-450	8	15k–17k	94k–107k	P_{MC}
Knott-3D-550	8	27k–31k	173k–195k	P_{MC}
Modularity Clustering	6	34–115	561–6555	P_{MC}
CREMI-small	3	20k–35k	170k–235k	P_{MC}
CREMI-large	3	430k–620k	3.2M–4.1M	P_{MC}
Fruit-Fly Level 1–4	4	5M–11M	28M–72M	P_{MC}
Fruit-Fly Global	1	90M	650M	P_{MC}
Ising Chain	30	100–300	4950–44850	P_{CUT}
2D Torus	9	100–400	200–800	P_{CUT}
3D Torus	9	125–343	375–1029	P_{CUT}
Deconvolution	2	1001	11k–34k	P_{CUT}
Super Resolution	2	5247	15k–25k	P_{CUT}
Texture Restoration	4	7k–22k	59k–195k	P_{CUT}

tissue. The *Fruit-Fly* instances were generated from volume image scans of fruit fly brain matter. The global problem is the largest instance in this study with roughly 650 million variables. It represents the current limit of what can be tackled by state-of-the-art local search algorithms. The instances *Level 1–4* are progressively simplified versions of the global problem obtained via block-wise domain decomposition (Pape et al., 2017).

For the max-cut problem we use two different types of instances. (i) The datasets *Ising Chain*, *2D Torus* and *3D Torus* contain instances that stem from applications in statistical physics Liers et al. (2005). The instances in *Ising Chain* assume a linear order on the nodes. For any pair of nodes there is an edge with an associated cost. The absolute values of the costs decrease exponentially with the distance of the nodes in the linear order. The instances in *2D Torus* and *3D Torus* are defined on toroidal grid graphs in two, resp. three dimensions with Gaussian distributed weights. (ii) The datasets *Deconvolution*, *Super Resolution* and *Texture Restoration* contain QPBO instances originating from image processing applications (Rother et al., 2007; Verma and Batra, 2012) that are converted to our formulation of the max-cut problem. The transformation introduces an additional node that is connected with all other nodes. A cut (uncut) edge to the additional node signifies label 0 (resp. 1). The instance size statistics for all data sets are summarized in Table 3.1.

Results In Table 3.2, we report for the multicut instances the average graph sizes after shrinking the instances with our algorithms from Section 3.5. In Figure 3.5, the contribu-

Table 3.2: For each dataset the table reports the average fraction of remaining nodes and edges after applying our method, respectively the method from Lange et al. (2018) (lower is better). †Results for *Fruit-Fly Global* are without ICP-based candidate subgraphs.

	Our		Lange et al. (2018)	
Data set	$\|V\|$	$\|E\|$	$\|V\|$	$\|E\|$
Image Seg.	27.7%	27.4%	63.7%	62.7%
Knott-3D-150	9.7%	9.6%	75.2%	88.3%
Knott-3D-300	54.8%	61.6%	76.7%	91.6%
Knott-3D-450	66.9%	77.6%	77.6%	92.4%
Knott-3D-550	67.8%	79.0%	77.8%	92.6%
Mod. Clustering	88.7%	80.6%	92.0%	85.1%
CREMI-small	33.8%	31.9%	76.6%	75.3%
CREMI-large	44.0%	44.2%	83.7%	86.6%
Fruit-Fly Level 1–4	8.7%	9.6%	24.6%	27.9%
Fruit-Fly Global†	56.3%	51.8%	77.9%	74.5%

tions of the individual persistency criteria are separated and compared. It can be seen from Table 3.2 and Figure 3.5 that our criteria enable finding substantially more persistent variables than the prior work (Alush and Goldberger, 2012; Lange et al., 2018). In relation to the graph sizes after shrinking with the baseline (Lange et al., 2018), our method achieves an additional size reduction of about 30–60% for the large *CREMI* and *Fruit-Fly* instances. This shows that our algorithms find persistent variable assignments that are harder to detect than with the criteria from prior work.

In Table 3.3 we report for the max-cut instances the average graph size reduction on each dataset. For the QPBO instances we compare to the QPBO method (Rother et al., 2007). For the original max-cut instances we are unaware of any baseline method and the QPBO method is not applicable. In Figure 3.6 we compare the contribution of the different subgraph criteria. It can be seen that our method solves all *Ising Chain* instances to optimality, which is facilitated by their particular distribution of the weights. On *2D Torus* we achieve substantial size reductions and on the denser *3D Torus* instances we find few persistencies. Our results on the QPBO instances are on a par with Rother et al. (2007) for *Deconvolution* and *Super Resolution* while our method is less effective for *Texture Restoration*.

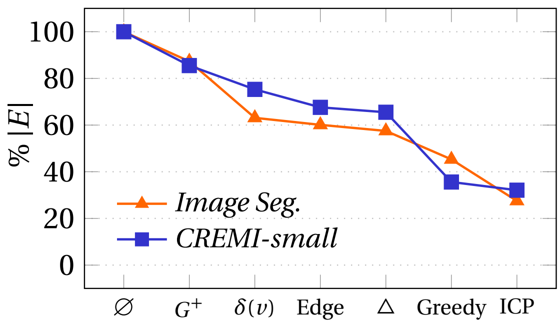

Figure 3.5: The figure shows the average fraction of remaining variables after shrinking the instance with progressively more expensive persistency criteria. The criteria added are from left to right: none [∅], connected components of G^+ [G^+], single node cuts [$\delta(v)$], edge subgraphs [Edge], triangle subgraphs [△], greedy subgraphs [Greedy], ICP candidate subgraphs and reduced cost fixing [ICP].

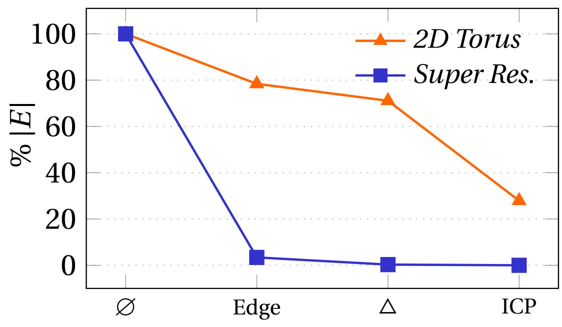

Figure 3.6: The figure shows the average fraction of remaining variables after shrinking the instance with progressively more expensive persistency criteria. The criteria added are from left to right: none [∅], edge subgraphs [Edge], triangle subgraphs [△], ICP candidate subgraphs and reduced cost fixing [ICP].

Table 3.3: For each dataset, the table reports the average fraction of remaining nodes and edges after applying our method, respectively the QPBO method (Rother et al., 2007) (lower is better). Note that the latter is not applicable to original max-cut instances due to symmetries.

	Our		Rother et al. (2007)	
Data set	$\|V\|$	$\|E\|$	$\|V\|$	$\|E\|$
Ising Chain	0.0%	0.0%	n/a	n/a
2D Torus	23.6%	27.9%	n/a	n/a
3D Torus	94.8%	98.1%	n/a	n/a
Deconvolution	61.0%	56.5%	61.0%	56.5%
Super Resolution	0.0%	0.0%	0.2%	0.1%
Texture Restoration	98.4%	98.5%	58.8%	57.3%

3.7 CONCLUSION

We have presented combinatorial persistency criteria for the multicut and max-cut problem. Moreover, we have devised efficient algorithms to check our criteria. For multicut our method achieves a substantial improvement over prior work when evaluated on common benchmarks as well as practical instances. For max-cut we are, to the best of our knowledge, the first to propose an algorithm that computes persistent variable assignments for the general problem. For the special case of QPBO problems, our method matches the performance of prior work on some instances. Our results demonstrate the feasibility of computing persistent variable assignments for NP-hard graph cut problems in practice. Besides acquiring partial optimality guarantees, our approach is a helpful tool for shrinking problem sizes and thus essential toward identifying globally optimal solutions.

4 TIGHTNESS OF THE CYCLE RELAXATION

4.1 INTRODUCTION AND RELATED WORK

In this chapter we study the quality of the cycle relaxation (P_{CYC}) of the multicut problem (P_{MC}) from a theoretical perspective. In contrast to Chapter 3, where we exploited the signs and absolute values of the edge costs for computing partial optimal solutions, this chapter is devoted to a study of the cycle relaxation purely based on the sign pattern of the cost vector. We pose the problem of characterizing those sign patterns for which the cycle relaxation is always tight, i.e. for which problem (P_{CYC}) has the same solution as (P_{MC}), regardless of the magnitude of the costs. This is equivalent to characterizing integrality of the conflicted cycle covering polyhedron associated with signed graphs.

The analogous question for the max-cut problem, which arises naturally when MC is replaced by CUT, has been considered in the literature (Grötschel and Pulleyblank, 1981; Guenin, 2001). The corresponding polyhedron is called *odd circuit covering polyhedron* and those signed graphs for which it is integral are called *weakly bipartite graphs*. Guenin (2001) characterized the weakly bipartite signed graphs in terms of forbidden signed minors, which is a seminal result in combinatorial optimization. It has been transferred by Weller (2016) and independently by Michini (2016) to the Boolean quadric polytope, where it implies a closely related signed minor condition for binary quadratic programming relaxations. For the multicut problem, we propose a similar approach in terms of *strong minors* of signed graphs (Cornaz, 2011).

The classical multicut formulation (Section 1.2.4) and its dual problem, the multi-commodity flow problem, have been widely studied in the literature (Schrijver, 2003). However, most of the research has been devoted to conditions under which the solutions of the fractional and integral maximum multi-commodity flow problem coincide. Cornaz (2011) asked when an exact min-max-relation between the minimum multicut and integral maximum multi-commodity flow holds. We, however, are interested in the case when the cycle relaxation of the multicut problem gives an exact solution, regardless of how the dual problem behaves. This question has been considered to lesser extent in the literature.

The LP relaxation of the classical multicut formulation is integral for up to two terminal pairs, which is a consequence of the well-known *max-flow-min-cut*-Theorem (Ford and Fulkerson, 1956), respectively a result due to Hu (1963) on two-commodity flows, cf. (Schrijver, 2003, Cor71.1d). These results directly carry over to the cycle relaxation (P_{CYC}) for up to two negative edges. Other special cases that do not directly carry over to our formulation are due to Tang (1964); Sakarovitch (1966); Rothfarb and Frisch (1969); Karzanov (1989). If G is series-parallel (it has no K_4 minor), then (P_{CYC}) is tight, since in this case CYC = MC (Chopra, 1994).

We characterize the integrality of (P_{CYC}) for the cases when the positive subgraph is either a tree or a circuit by identifying two classes of forbidden substructures. While in the former case the multicut problem (P_{MC}) remains NP-hard (Garg et al., 1997), in the latter case it can be solved in polynomial time (Bentz et al., 2009). For general graphs, we present another forbidden substructure and establish a connection to open problems in the general theory of ideal clutters.

This chapter builds upon results in the theory of ideal clutters (Lehman, 1979, 1990; Seymour, 1990; Cornuéjols and Novick, 1994), in particular Lehman's theorem, which is a key ingredient towards proving integrality of LP relaxations of covering problems. Other works that apply Lehman's theorem include (Guenin, 2001; Schrijver, 2002; Abdi et al., 2016).

4.2 FLOW COVERS

In this chapter we consider signed graphs $G = (V, E^+, E^-)$, where E^+ and E^- denote the disjoint sets of positive, respectively negative edges. Recall that the sign pattern of $\theta \in \mathbb{R}^E$ defined on the edge set E of a graph induces a signature. By Lemma 1.5, integrality of the cycle relaxation (P_{CYC}) for a given signed graph $G = (V, E^+, E^-)$ is equivalent to integrality of the conflicted cycle covering formulation (1.6). For convenience and to accord with (Cornaz, 2011), in this chapter we refer to any conflicted cycle as *flow*. Moreover, we denote by $\mathscr{F}$ the set of flows of the signed graph G. Accordingly, the *flow covering relaxation* of the multicut problem (up to a constant) reads

$$\begin{aligned} \min \quad & \sum_{e \in E} |\theta| x_e & & \\ \text{s.t.} \quad & \sum_{e \in C} x_e \geq 1 & \forall C \in \mathscr{F} & \qquad (4.1) \\ & x_e \geq 0 & \forall e \in E = E^+ \cup E^-. & \qquad (4.2) \end{aligned}$$

Therefore, it is easy to see that the cycle relaxation (P_{CYC}) is integral for a given signed graph if, and only if, the associated flow covering polyhedron is integral.

Corollary 4.1. *Let $G = (V, E^+, E^-)$ be a signed graph and let $E = E^+ \cup E^-$. It holds that the solutions of (P_{MC}) and (P_{CYC}) coincide for every $\theta \colon E \to \mathbb{R} \setminus \{0\}$ whose sign pattern agrees with the signature of G if, and only if, the polyhedron defined by (4.1) – (4.2) is integral.*

4.3 IDEAL CLUTTERS

In this section we introduce the necessary terminology from the theory of ideal clutters and define the flow clutter.

With any 0-1-matrix A we associate the (fractional) covering polyhedron

$$\mathrm{P}_A = \{x \geq 0 \mid Ax \geq \mathbb{1}\}. \qquad (4.3)$$

The 0-1-matrix A is called *ideal* if P_A is integral.

A family of subsets $\mathscr{C}$ of a finite ground set $E(\mathscr{C})$ is called *clutter* if no member of $\mathscr{C}$ is contained in another. With any clutter $\mathscr{C}$ we naturally identify the 0-1-matrix $A(\mathscr{C})$ whose rows are the characteristic vectors of the members of $\mathscr{C}$. By definition, no row vector of $A(\mathscr{C})$ dominates another, thus the constraint system that defines $\mathsf{P}_{A(\mathscr{C})}$ is irredundant. A clutter $\mathscr{C}$ is called *ideal* if the associated matrix $A(\mathscr{C})$ is ideal and otherwise it is called *non-ideal*.

A *cover* of $\mathscr{C}$ is a subset $B \subseteq E(\mathscr{C})$ such that for all $C \in \mathscr{C}$ it holds that $B \cap C \neq \emptyset$. The *blocker* of $\mathscr{C}$ is the clutter that is formed by the minimal covers of $\mathscr{C}$.

The *contraction* of an element $e \in E(\mathscr{C})$ gives a clutter $\mathscr{C}/e$ over the ground set $E(\mathscr{C}) \setminus \{e\}$ that consists of the minimal sets from $\{C \setminus \{e\} \mid C \in \mathscr{C}\}$. The *deletion* of an element $e \in E(\mathscr{C})$ gives the clutter $\mathscr{C} \setminus e = \{C \in \mathscr{C} \mid e \notin C\}$ over the ground set $E(\mathscr{C}) \setminus \{e\}$. A *minor* of $\mathscr{C}$ is any clutter obtained from $\mathscr{C}$ by a series of contraction and deletion operations. Contraction and deletion operations are commutative. The minor operations on clutters correspond naturally to restricting the polyhedron $\mathsf{P}_{A(\mathscr{C})}$ by setting variables to 0 or 1. More precisely, it holds that

$$P_{A(\mathscr{C}/e)} = P_{A(\mathscr{C})} \cap \{x \mid x_e = 0\},$$
$$P_{A(\mathscr{C}\setminus e)} = P_{A(\mathscr{C})} \cap \{x \mid x_e = 1\}.$$

The property of idealness is closed under taking minors (Seymour, 1977).

A clutter is called *minimally non-ideal (MNI)* if it is not ideal but any (proper) minor is ideal. Clearly, a clutter is ideal if, and only if, it has no MNI minor.

Example 4.1. For any $2 \leq k \in \mathbb{N}$, the clutter $\mathscr{D} = \big\{\{1,\dots,k\},\{0,1\},\dots,\{0,k\}\big\}$ over the ground set $\{0,1,\dots,k\}$ is called *degenerate projective plane* of order k. Any degenerate projective plane is MNI (Lehman, 1979).

Lehman's Theorem For any MNI clutter $\mathscr{C}$, the *core* of $\mathscr{C}$, denoted by $\overline{\mathscr{C}}$, is the clutter of minimum size members of $\mathscr{C}$. Lehman (1990) proved an important characterization of MNI clutters in terms of their cores. We state Lehman's theorem below for later reference, an accessible proof can also be found in (Seymour, 1990).

Theorem 4.1 (Lehman (1990))**.** *Let $\mathscr{C}$ be an MNI clutter that is not a degenerate projective plane and let $\mathscr{B}$ be its blocker. Then both $\overline{\mathscr{C}} = \{C_i\}_i$ and $\overline{\mathscr{B}} = \{B_i\}_i$ consist of $n = E(\mathscr{C})$ members and can be ordered suitably such that for some $c, b \in \mathbb{N}$ it holds that*

(i) $cb \geq n+1$

(ii) $\forall C \in \overline{\mathscr{C}} : |C| = c$ *and* $\forall B \in \overline{\mathscr{B}} : |B| = b$

(iii) $\forall e \in E(\mathscr{C}) : |\{C \in \overline{\mathscr{C}} \mid e \in C\}| = c$ *and* $|\{B \in \overline{\mathscr{B}} \mid e \in B\}| = b$

(iv) $\forall i, j \in [n] : |C_i \cap B_j| = \begin{cases} cb - n + 1 & \text{if } i = j \\ 1 & \text{else} \end{cases}$

(v) $\forall e, f \in E(\mathscr{C}) : |\{i \in [n] : e \in C_i, f \in B_i\}| = \begin{cases} cb - n + 1 & \text{if } e = f \\ 1 & \text{else.} \end{cases}$

In particular, the polyhedron $\mathsf{P}_{A(\mathscr{C})}$ has the unique fractional vertex $\frac{1}{c}\mathbb{1}$.

4.3.1 The Flow Clutter

Apparently, the set of flows $\mathscr{F}$ of a signed graph $G=(V,E^+,E^-)$, identified by their edge sets, is a clutter, which we call the *flow clutter*. We define a *strong minor* (Cornaz, 2011) of $\mathscr{F}$ as a minor (without singleton elements) that is obtained by a series of contractions of positive edges and deletions of arbitrary edges. A *strong minor* of the signed graph G is a minor (without self-loops) that is obtained, analogously, by contraction of positive edges only and deletion of arbitrary edges. Clearly, the strong minors are those minors that correspond to flow clutters defined by signed graphs.

Corollary 4.2. *Let $G=(V,E^+,E^-)$ be a signed graph and $\mathscr{F}$ its flow clutter. Then any strong minor $\mathscr{F}'$ of $\mathscr{F}$ corresponds to a strong minor H of G and vice versa.*

We further call a flow clutter *weakly MNI* if it is not ideal, but any (proper) strong minor is ideal. It is clear that any MNI clutter is also weakly MNI. However, weakly MNI clutters may have MNI minors that are constructed by contraction of negative edges.

Lemma 4.1. *For any flow clutter $\mathscr{F}$ the following are equivalent:*

(i) $\mathscr{F}$ is ideal

(ii) $\mathscr{F}$ has no weakly MNI strong minor

Proof. Every weakly MNI strong minor has an MNI minor, so if $\mathscr{F}$ is ideal, then it cannot have a weakly MNI strong minor. Conversely, suppose $\mathscr{F}$ has no weakly MNI strong minor, but some MNI minor $\mathscr{F}'$. Then, since the contraction and deletion operations are commutative, this minor is obtained from a strong minor $\mathscr{F}''$ by a series of contractions of negative edges. Thus, $\mathscr{F}''$ is weakly MNI, which is a contradiction. □

Lemma 4.2. *Let $\mathscr{F}$ be weakly MNI. Then G has no parallel edges.*

Proof. Let $f,g\in E$ be a pair of parallel edges. First assume that $f,g\in E^-$ are both negative. Let x be any vertex of $\mathsf{P}_{A(\mathscr{F})}$. If $x_f<x_g$, then $x=1/2(y+z)$, where $y,z\in \mathsf{P}_{A(\mathscr{F})}$ agree with x except for $y_g=x_f$ and $z_g=2x_g-x_f$. Thus, it must hold that $x_f=x_g$. Now suppose x is a fractional vertex of $\mathsf{P}_{A(\mathscr{F})}$. Let $x_{\setminus f}\in \mathsf{P}_{A(\mathscr{F}\setminus f)}$ denote the vector obtained from x by setting $x_f=1$. Since $\mathscr{F}$ is weakly MNI, $x_{\setminus f}$ can be written as a convex combination of 0-1-vectors $y^k\in \mathsf{P}_{A(\mathscr{F}\setminus f)}$. Every y^k is a vertex, so $y^k_g=y^k_f=1$. This implies that $x_f=x_g=1$, which is a contradiction.

If $f,g\in E^+$ are both positive, then the proof is completely analogous. Otherwise (w.l.o.g.) $f\in E^+$ and $g\in E^-$, so the cycle induced by f and g is a flow. Let x be any vertex of $\mathsf{P}_{A(\mathscr{F})}$. We have $x_f+x_g\geq 1$. Assume that $x_f+x_g>1$. Since x is a vertex, neither x_f nor x_g can be decreased without violating an inequality of $\mathsf{P}_{A(\mathscr{F})}$. Therefore, there exists a flow C containing f and a path P that induces a flow with g such that $x_P+x_g=1$ and $x_{C\setminus\{f\}}+x_f=1$. Further, the set $P\cup C\setminus\{f\}$ contains a flow. This implies that

$$1\leq x_{C\setminus\{f\}}+x_P=2-x_f-x_g \tag{4.4}$$

$$\implies x_f+x_g\leq 1, \tag{4.5}$$

which is a contradiction. Hence, it holds that $x_f + x_g = 1$. Now suppose x is a fractional vertex of $P_{A(\mathscr{F})}$. Since $\mathscr{F}$ is weakly MNI, the vector $x_{\setminus f}$ can be written as a convex combination of 0-1-vectors $y^k \in \mathsf{P}_{A(\mathscr{F}\setminus f)}$. Every y^k is a vertex, so they satisfy $y_f^k = 1$ and $y_g^k = 0$. This implies that also $x_g = 0$ and $x_f = 1$, which is a contradiction. □

It is straightforward to see that flow clutters and minors of flow clutters obtained by contraction of E^- do not correspond to degenerate projective planes.

Lemma 4.3. *There is no degenerate projective plane that corresponds to a flow clutter. Similarly, no degenerate projective plane of order $k \geq 3$ corresponds to a clutter that is obtained from a flow clutter by contraction of all negative edges (purely as members of the clutter).*

Proof. Let $\mathscr{D} = \big\{\{1,\dots,k\},\{0,1\},\dots,\{0,k\}\big\}$ be a degenerate projective plane of order k. If $0 \in E^-$, then $1,\dots,k \in E^+$, which implies that the first member of $\mathscr{D}$ has no negative edge. If $0 \notin E^+$, then $1,\dots,k \in E^-$, which implies that the first member of $\mathscr{D}$ has no positive edge.

Now, suppose $\mathscr{D}$ consists of the edge sets of the positive paths of a flow clutter. Since $\{1,\dots,k\}$ is a path (and not a cycle) of length $k \geq 3$ one of the members $\{0,1\},\dots,\{0,k\}$ cannot represent a connected path. □

Lemma 4.4. *Let $\mathscr{F}$ be weakly MNI and suppose x is a fractional vertex of $\mathsf{P}_{A(\mathscr{F})}$. Then $0 < x_e < 1$ for all $e \in E^+$ and $x_e < 1$ for all $e \in E^-$.*

Proof. Every vertex of $\mathsf{P}_{A(\mathscr{F})}$ satisfies $0 \leq x_e \leq 1$ for all $e \in E$. If $x_e = 1$ for any $e \in E$, then delete e to obtain a strong minor of $\mathscr{F}$ that is non-ideal. If $x_e = 0$ for $e \in E^+$, then contract e to obtain a strong minor of $\mathscr{F}$ that is non-ideal. □

For any non-ideal flow clutter $\mathscr{F}$ and any fractional vertex x of $\mathsf{P}_{A(\mathscr{F})}$, let

$$E_0^-(x) = \{e \in E^- \mid x_e = 0\}$$

denote the subset of negative edges where x takes the value 0.

Lemma 4.5. *Let $\mathscr{F}$ be weakly MNI. Then there exists a fractional vertex x of $\mathsf{P}_{A(\mathscr{F})}$ such that the clutter $\mathscr{F}/E_0^-(x)$ obtained by contraction of $E_0^-(x)$ is MNI. In particular, if x is a fractional vertex of $\mathsf{P}_{A(\mathscr{F})}$ and $E_0^-(x) = E^-$, then $\mathscr{F}/E^-$ is MNI.*

Proof. Take some fractional vertex x of $\mathsf{P}_{A(\mathscr{F})}$ and suppose the clutter $\mathscr{C} = \mathscr{F}/E_0^-(x)$ is not MNI. If there exists an element $e \in E^+ \cup E^- \setminus E_0^-(x)$ such that $\mathscr{C}\setminus e$ is non-ideal, then, since $\mathscr{C}\setminus e = (\mathscr{F}\setminus e)/E_0^-(x)$, it follows that $\mathscr{F}\setminus e$ is non-ideal, so $\mathscr{F}$ is not weakly MNI. A similar argument shows that there cannot be any $e \in E^+$ such that $\mathscr{C}/e$ is non-ideal. Thus, there exists some $f \in E^- \setminus E_0^-(x)$ such that $\mathscr{C}/f$ is non-ideal. Hence, $P_{A(\mathscr{F})}$ has a fractional vertex y with $y_e = 0$ for all $e \in E_0^-(x)$ and $y_f = 0$. Replacing x by y and repeating this argument eventually yields the desired x, as E^- is finite. The argument further shows that if $E_0^-(x) = E^-$, then $\mathscr{F}/E^-$ must already be MNI. □

Let $A^+(\mathscr{F})$ denote the submatrix of $A(\mathscr{F})$ that consists of the columns associated with E^+ and define $A^-(\mathscr{F})$ similarly.

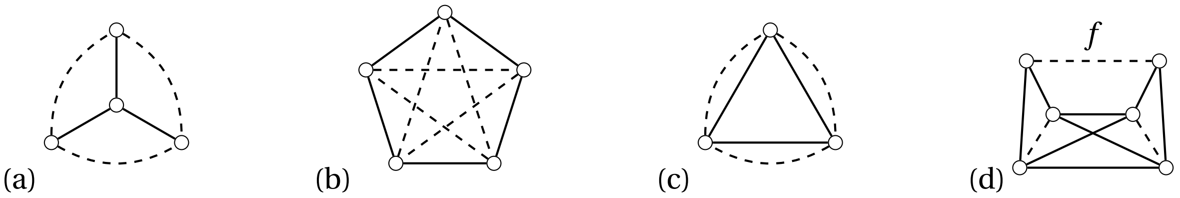

Figure 4.1: Dashed lines depict negative edges, solid lines depict positive edges. (a) An odd flow-star with three positive edges. (b) The smallest non-ideal odd flow-circuit. (c) The only ideal odd flow-circuit. (d) Flow-split-K_5 with fractional vertex x such that $x_f = 0$ and $x_e = 1/3$ for all $e \neq f$.

4.4 IDEALNESS OF FLOW CLUTTERS

In this section we discuss the idealness of flow clutters. We first employ the stronger notion of balancedness to characterize an instructive special case.

A square 0-1-matrix A is called *2-circulant* if there exist appropriate permutations of the rows and columns such that the permuted matrix $\tilde{A}$ is of the form

$$\tilde{A} = \begin{pmatrix} 1 & 1 & & & \\ & 1 & 1 & & \\ & & \ddots & \ddots & \\ & & & 1 & 1 \\ 1 & & & & 1 \end{pmatrix}.$$

A 0-1-matrix A is called *balanced* if it does not contain any 2-circulant submatrix of odd size. Balanced matrices were introduced by Berge (1972), who also proved the following result.

Theorem 4.2 (Berge (1972)). *Any balanced 0-1-matrix is ideal.*

4.4.1 Positive Trees

We first consider the case that $G^+ = (V, E^+)$ is a tree. Note that in this case the multicut problem remains NP-hard (Garg et al., 1997).

A *flow-star* is a signed graph $S = (V_S, E_S^+, E_S^-)$ on $V_S = \{v_0, v_1, \dots, v_k\}$, where $k \geq 3$, such that $E_S^+ = \{v_0 v_i \mid 1 \leq i \leq k\}$ and $E_S^- = \{v_i v_{i+1} \mid 1 \leq i \leq k-1\} \cup \{v_k v_1\}$. A flow-star is called odd if k is odd. See Figure 4.1 (a) for an example. Any odd flow-star S defines a non-ideal flow clutter as the vector x defined by $x_e = 1/2$ for $e \in E_S^+$ and $x_e = 0$ for $e \in E_S^-$ is a fractional vertex of the associated polyhedron.

Lemma 4.6. *Let $G^+ = (V, E^+)$ be a tree. If G has no odd flow-star strong minor, then $A(\mathscr{F})$ is balanced.*

Proof. For contraposition, assume $A(\mathscr{F})$ is not balanced, so it has a 2-circulant submatrix B of odd order. Note that, since G^+ is a tree, every negative edge induces exactly one flow.

Thus, the columns of B correspond to positive edges only. We construct an odd flow-star strong minor of G. First, delete all edges that do not correspond to any flow associated with the rows of B. Then, contract all positive edges from G that do not correspond to any column of B. This yields a minor $S = (V_S, E_S^+, E_S^-)$ of G. Clearly, since G^+ is a tree, S^+ is also a tree. Further, we never contract any parallel edge which shows that S is a strong minor. The structure of B implies that the graph S has an odd number of flows of length three. The flows can be cyclically ordered such that every adjacent pair of flows shares a positive edge. Hence, it follows that S must be an odd flow-star. □

Corollary 4.3. *Let $G^+ = (V, E^+)$ be a tree. Then $\mathscr{F}$ is ideal if and only if G has no odd flow-star strong minor.*

Proof. If G has no odd flow-star strong minor, then $\mathscr{F}$ is ideal by Lemma 4.6 and Theorem 4.2. Conversely, if G has an odd flow-star strong minor, then $\mathscr{F}$ cannot be ideal, since any odd flow-star is non-ideal and idealness is preserved under taking minors. □

4.4.2 Positive Circuits

Now we consider the case that $G^+ = (V, E^+)$ is a circuit. For weakly MNI flow clutters on positive circuits, we can show that the minor obtained from contraction of all negative elements is MNI.

Lemma 4.7. *Let G^+ be a circuit and suppose that $\mathscr{F}$ is weakly MNI. There exists a fractional vertex x of $\mathsf{P}_{A(\mathscr{F})}$ such that $E_0^-(x) = E^-$.*

Proof. Let x be a fractional vertex of $\mathsf{P}_{A(\mathscr{F})}$ such that $\mathscr{F}/E_0^-(x)$ is MNI and assume that $0 < x_f < 1$ for some $f \in E^-$. If $\mathscr{F}/E_0^-(x)$ is a degenerate projective plane of order $k \geq 2$, then it has a member of size two that contains f. Thus, f is parallel to some positive edge, which is a contradiction to Lemma 4.2. Hence, the $\mathscr{F}/E_0^-(x)$ is not a degenerate projective plane and, by Theorem 4.1, the flows that share f have size two, since f induces exactly two flows. This implies that G^+ has size two and therefore $\mathscr{F}$ is ideal, which is a contradiction. □

A *flow-circuit* is a signed graph $C = (V_C, E_C^+, E_C^-)$ on $V_C = \{v_1, \ldots, v_k\}$, where $k \geq 3$, such that $E_C^+ = \{v_i v_{i+1} \mid 1 \leq i \leq k-1\} \cup \{v_k v_1\}$ and $E_C^- = \{v_i v_{i+2} \mid 1 \leq i \leq k-2\} \cup \{v_{k-1} v_1, v_k v_1\}$. A flow-circuit is called odd if k is odd. Any odd flow-circuit C with $|E_C^+| \geq 5$ defines a non-ideal flow clutter as the vector x defined by $x_e = 1/2$ for $e \in E_C^+$ and $x_e = 0$ for $e \in E_C^-$ is a fractional vertex of the associated polyhedron. See Figure 4.1 (b) for an example. By another application of Lehman's Theorem we can characterize idealness of $\mathscr{F}$ by forbidden odd flow-circuit strong minors as follows.

Theorem 4.3. *Let G^+ be a circuit. Then the clutter $\mathscr{F}$ is ideal if and only if G has no odd flow-circuit strong minor $C = (V_C, E_C^+, E_C^-)$ with $|E_C^+| \geq 5$.*

Proof. If G has an odd flow-circuit strong minor C with $|E_C^+| \geq 5$, then $\mathscr{F}$ cannot be ideal as idealness is preserved under taking minors.

Conversely, suppose $\mathscr{F}$ is non-ideal and weakly MNI. Consider the minor $\mathscr{P} = \mathscr{F}/E^-$ obtained by contraction of all negative elements in $\mathscr{F}$. Since $\mathscr{F}$ is weakly MNI, the clutter $\mathscr{P}$

is MNI (Lemmas 4.5 and 4.7) and consists of the edge sets of paths in G^+ associated with the flows in $\mathscr{F}$. As $\mathscr{P}$ is not a degenerate projective plane (Lemma 4.3), Theorem 4.1 implies that there are $n = |E^+|$ minimum members of $\overline{\mathscr{P}}$ that all have some constant length p. Let $\mathscr{M}$ be the blocker of $\mathscr{P}$ so the n members of $\overline{\mathscr{M}}$ have some constant size m. By Theorem 4.1, it holds that $pm \geq n+1$. Further, we have $2 \leq p \leq \frac{n}{2}$ as G^+ is a circuit and the members of $\overline{\mathscr{P}}$ have minimum length. As the n paths in $\overline{\mathscr{P}}$ of length p can be arranged cyclically, it is apparent that $m = \lceil \frac{n}{p} \rceil$. Now, if $p = \frac{n}{2}$, then $m = 2$ and thus $pm = n < n+1$. Therefore, we must have that $p \leq \frac{n-1}{2}$. In particular, every negative edge $f \in E^-$ corresponds to exactly one path $P_f \in \overline{\mathscr{P}}$ and $\overline{\mathscr{P}} = \mathscr{P}$.

Assume that $p \geq 3$. We show that this leads to a contradiction and thus $p = 2$. Take some $M \in \overline{\mathscr{M}}$. By Theorem 4.1, there exists a unique $P_f \in \mathscr{P}$ for $f \in E^-$ such that $|P_f \cap M| = pm - n + 1$ and $|P \cap M| = 1$ for all $P \in \mathscr{P}$ with $P \neq P_f$. We define a vector $x \in \mathbb{R}^{E^+ \cup E^-}$ by

$$x_e = \begin{cases} \frac{1}{p-1} & e \in E^+ \setminus M \\ 0 & e \in M \\ \frac{pm-n}{p-1} & e = f \\ 0 & e \in E^- \setminus f. \end{cases} \tag{4.6}$$

The vector $x \geq 0$ is constructed such that for every $P \in \mathscr{P}$, the corresponding flow covering inequality of $\mathsf{P}_{A(\mathscr{F})}$ is tight. Indeed, it holds that

$$x(P_f \cup \{f\}) = \frac{p - pm + n - 1}{p-1} + \frac{pm - n}{p-1} = \frac{p-1}{p-1} = 1 \tag{4.7}$$

and, for all $e \in E^-$ with $e \neq f$ that

$$x(P_e \cup \{e\}) = \frac{p-1}{p-1} + 0 = 1. \tag{4.8}$$

Feasibility of x is clear, since each path in E^+ that corresponds to a flow but is not of minimum size contains at least two distinct members of $\mathscr{P}$. Therefore, it holds that x is a vertex of $\mathsf{P}_{A(\mathscr{F})}$. Further, since $p \geq 3$, the vertex x is fractional, which implies that $\mathscr{F}$ cannot be weakly MNI. Thus, it must hold that $p = 2$.

This shows that, as $\mathscr{P}$ is MNI, the matrix $A(\mathscr{P})$ must be an odd 2-circulant matrix. Moreover, it holds that $|E^+| \geq 5$ as otherwise $\mathscr{F}$ would be ideal. Hence, the signed graph G is an odd flow-circuit with $|E^+| \geq 5$. □

4.4.3 General Graphs

In the case of positive trees and positive circuits we exploited the fact that for weakly MNI flow clutters $\mathscr{F}$, any fractional vertex x of $\mathsf{P}_{A(\mathscr{F})}$ such that $\mathscr{F} / E_0^-(x)$ is MNI satisfies $E_0^-(x) = E^-$. In this section we show for general graphs, if $E_0^-(x) \neq E^-$, then the corresponding MNI clutter has a *fat* core.

The core $\overline{\mathscr{C}}$ of an MNI clutter $\mathscr{C}$ is called *fat* if $cb - n + 1 \geq 3$, where c, b and n are the constants from Theorem 4.1. Currently, only three distinct fat cores of MNI clutters are known (Cornuéjols et al., 2009), the clutter F_7 of lines of the Fano plane (which is its own blocker), the clutter $\tau(K_5)$ of triangles of K_5 and its blocker. In each case it holds that $cb - n + 1 = 3$.

Lemma 4.8. *Let $\mathscr{F}$ be a weakly MNI flow clutter. Let x be a fractional vertex of $\mathsf{P}_{A(\mathscr{F})}$ such that $\mathscr{C} = \mathscr{F}/E_0^-(x)$ is MNI and $E_0^-(x) \neq E^-$. Then the core $\overline{\mathscr{C}}$ is fat.*

Proof. Since $E_0^-(x) \neq E^-$ there exists some $f \in E^-$ such that $0 < x_f < 1$. Assume that $\mathscr{C}$ is a degenerate projective plane of order $k \geq 2$. Then f is parallel to some positive edge, which is a contradiction to $\mathscr{F}$ being weakly MNI.

Thus, the clutter $\mathscr{C}$ is not a degenerate projective plane. Let $\overline{\mathscr{C}} = \{C_i\}_i$ and $\overline{\mathscr{B}} = \{B_i\}_i$ denote the cores of $\mathscr{C}$ and its blocker $\mathscr{B}$. By Theorem 4.1, there are $i, j \in [n]$ with $i \neq j$ such that $f \in C_i \cap B_i$ and $f \in C_j \cap B_j$. Further, it holds that $C_i \cap C_j = \{f\}$. Therefore, the set of edges $(C_i \cup C_j) \setminus \{f\}$ induces a positive cycle. Since $|C_i \cap B_i| = cb - n + 1 \geq 2$, there exists some $g \in E^+$ such that $g \in C_i \cap B_i$. There is another flow $C_k \in \overline{\mathscr{C}}$, $k \neq i, j$ such that $C_i \cap C_k = \{g\}$. Now, since $(C_i \cup C_j) \setminus \{f\}$ is a positive cycle, the set $(C_k \cup C_i \cup C_j) \setminus \{f, g\}$ contains a flow, which must be covered by B_i. Hence, there exists another edge $h \in (C_i \cup C_j) \setminus \{f, g\} \cap B_i$. It must hold that $h \in C_i$ as $C_j \cap B_i = \{f\}$. This shows that $\{f, g, h\} \subseteq C_i \cap B_i$ and thus $cb - n + 1 \geq 3$. □

The instance depicted in Figure 4.1 (d), which we call *flow-split-K_5*, is weakly MNI. Let $\mathscr{F}$ be the corresponding flow clutter. It has an associated fractional vertex x such that $x_f = 0$ and $x_e = 1/3$ for all $e \neq f$. Further, the core of the MNI minor $\mathscr{F}/f$ is isomorphic to $\tau(K_5)$. To see this, take K_5, split an arbitrary vertex into two vertices with two neighbors each, connect them by f and sign the resulting graph appropriately. The other known fat cores of MNI clutters do not arise from flow clutters, as we point out in the following lemmas. This shows that any weakly MNI flow clutter that arises from a vertex x of $\mathsf{P}_{A(\mathscr{F})}$ with $E_0^-(x) \neq E^-$ and is different from the flow-split-K_5 would imply the existence of an unknown MNI clutter with a fat core.

Lemma 4.9. *Let $\mathscr{F}$ be a weakly MNI flow clutter and x be a fractional vertex of $\mathsf{P}_{A(\mathscr{F})}$ such that the clutter $\mathscr{C} = \mathscr{F}/E_0^-(x)$ is MNI. If $E_0^-(x) \neq E^-$, then $\mathscr{C}$ is not isomorphic to the blocker of $\tau(K_5)$.*

Proof. Assume that $\mathscr{C}$ is isomorphic to the blocker of $\tau(K_5)$. Then $\overline{\mathscr{C}}$ has 10 members with 4 elements each. By assumption, there is some $f \in E^-$ that is contained in 4 members of $\mathscr{C}$. Since all members are of minimum size and no edges may be parallel, at least 8 positive edges are needed to form the flows that include f and they are arranged as in Figure 4.1 (d) (without the other two negative edges). Furthermore, it is clear that $\overline{\mathscr{C}}$ cannot contain this substructure, as its members are composed of only 10 distinct elements. □

Lemma 4.10. *Let $\mathscr{F}$ be a weakly MNI flow clutter and x be a fractional vertex of $\mathsf{P}_{A(\mathscr{F})}$ such that the clutter $\mathscr{C} = \mathscr{F}/E_0^-(x)$ is MNI. If $E_0^-(x) \neq E^-$, then $\mathscr{C}$ is not isomorphic to the clutter F_7.*

Proof. We prove the assertion by contraposition. Assume $\mathscr{C}$ is isomorphic to F_7 and take some $f \in E^- \setminus E_0^-(x)$. As $x_f > 0$, one element of F_7 corresponds to f. Thus, since for any other element e of F_7 there is a member of F_7 that contains both e and f, it holds that $e \in E^+$. It follows that G^+ consists of three edge-disjoint paths of length two connecting the endpoints of f. Moreover, there is another member of F_7 that contains one edge from each such path and forms a path itself. This is impossible and thus the edge f cannot exist. □

4.5 DISCUSSION

In the proof for positive circuits, we consider the MNI minor of the flow clutter that consists of the corresponding positive paths. In fact, for both positive trees and positive paths we exploit the one-to-one correspondence between flows and its positive subpaths. Therefore it may seem more useful to consider the clutter of positive paths to start with and instead characterize the MNI minors of that clutter. After all, that approach is closer to the classical multicut formulation, where one seeks to cover a set of terminal paths. In this section, we discuss this issue by presenting two arguments why our approach is favorable nonetheless.

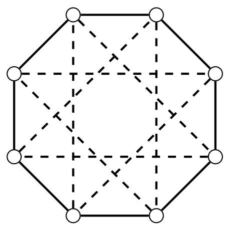

Figure 4.2: Dashed lines depict negative edges, solid lines depict positive edges. The depicted instance is not weakly MNI, but has an odd flow-circuit strong minor. However, if the negative edges are interpreted as terminal pairs instead, then the corresponding clutter of terminal paths is already MNI.

i. There is a direct correspondence between strong minors of flow clutters and (strong) signed graph minors that represent multicut instances. If instead, we consider the clutter of terminal paths, then the minors of that clutter are not in direct correspondence with the minors of the associated graph, due to the fact that the graph minors do not carry any information about the terminal pairs.

ii. There are fewer forbidden minors that need to be considered. A simple example is the flow-split-K_5 as depicted in Figure 4.1. There is not a unique MNI instance of the classical formulation that corresponds to the flow-split-K_5, since its negative edges with fractional entries can be replaced by positive edges and terminal pairs in two different ways (cf. Figure 1.2). Another (more sophisticated) example concerns instances defined on (positive) circuits. Note that the odd flow-stars and odd flow-circuits are instances of so-called *circulant* clutters, which can be represented by circulant 0-1-matrices. In fact, both structures correspond to 2-circulant matrices. Cornuéjols and Novick (1994) provide a complete list of all other MNI circulant clutters. If we consider the clutter of terminal paths defined on a circuit, then it turns out that it does not suffice to only forbid minors that correspond to 2-circulant clutters. For example, consider the instance depicted in Figure 4.2. It is not weakly MNI, but non-ideal as it has an odd flow-circuit strong minor. If instead, the negative edges are regarded as terminal pairs, then the corresponding clutter of terminal paths is the MNI circulant clutter $\mathscr{C}_8^3$, which consists of 8 members with 3 elements each.

4.6 CONCLUSION

We posed the problem of characterizing the tightness of the cycle relaxation of the multicut problem (P_{MC}) in terms of the sign pattern of the cost vector. It is equivalent to characterizing the integrality of the flow covering polyhedron for a signed graph, which is an analogue to the characterization of weakly bipartite signed graphs in connection to the max-cut problem. We present two infinite classes and an additional third instance of strong minors that are necessarily forbidden to guarantee an integral polyhedron associated with the flow clutter. Our results provide a characterization for the special cases when the positive subgraph is either a tree or a circuit. For general graphs we establish a connection between the characterization of ideal flow clutters and open problems in the general theory of ideal clutters.

Part II

Geometry of Lifted Multicuts

5 LIFTED MULTICUTS

5.1 INTRODUCTION

As an encoding of graph partitions, multicuts make explicit, for every pair of neighboring nodes, whether or not the nodes are in distinct components or not. To make explicit also for some non-neighboring nodes, whether or not the nodes are in distinct components, we define a procedure that lifts the multicuts of a base graph to the multicuts of a super graph that contains additional edges. This lifting is well-defined if we exploit the one-to-one correspondence between multicuts and partitions into induced subgraphs, which we call *decompositions*. The lifted multicuts are still in one-to-one correspondence with the decompositions of the base graph. Yet, they are a more expressive model of graph partitions as they encode connectedness not only locally, but also globally. The associated lifted multicut problem generalizes the multicut problem by incorporating non-local costs into the objective function.

In this work, we study the convex hull of characteristic vectors of lifted multicuts, the so-called lifted multicut polytope. We investigate conditions under which its fundamental inequalities are facet-defining. In particular, our work generalizes the characterization of facet-defining cycle inequalities for the multicut polytope due to Chopra and Rao (1993). Further, we define efficient separation procedures and apply these in a branch-and-cut algorithm to solve the lifted multicut problem.

Nowozin and Lampert (2010) study connectivity potentials for random field models and polyhedral relaxations. Generalizations of the multicut problem to multilinear objective functions are studied by Kim et al. (2014) and Kappes et al. (2016a). Applications of the lifted multicut problem and experimental comparisons to the correlation clustering problem in the field of computer vision are by Keuper et al. (2015b) and Tang et al. (2017) who find feasible solutions by local search (Keuper et al., 2015b), and by Beier et al. (2017) who find feasible solutions by consensus optimization (Beier et al., 2016). Keuper (2017) further applies a higher-order lifted multicut model to the task of motion segmentation.

5.2 LIFTING OF MULTICUTS

As before, we consider connected graphs $G = (V, E)$. In this section we introduce the lifting of multicuts of G to a super graph $\widehat{G} = (V, E \cup F)$ as well as the associated lifted multicut problem. To this end, we constrain the set of considered graph partitions of G so as to establish a bijection to its multicuts (cf. Figure 1.1).

Definition 5.1. Let $\Pi = \{U_1, \dots, U_k\}$ be a graph partition of $G = (V, E)$. If every U_i induces a connected component of G, then Π is called a *graph decomposition*.

Let $\mathscr{P}(V)$ denote the set of all partitions of V. Consider the map

$$\begin{aligned}\phi_G \colon \mathscr{P}(V) &\to 2^E \\ \Pi &\mapsto M\end{aligned}$$

defined by

$$uv \in \phi_G(\Pi) \iff \forall U \in \Pi : u \notin U \vee v \notin U. \tag{5.1}$$

Lemma 5.1. *The map ϕ_G defined by* (5.1) *is a bijection between the decompositions of G and the multicuts of G.*

Proof. First, we show that for any decomposition Π, the image $\phi_G(\Pi)$ is a multicut of G. Assume the contrary, i.e. there exists a cycle C of G such that $\phi_G(\Pi) \cap C = \{uv\}$ for some $uv \in E$. Then for all $U \in \Pi$ it holds that $u \notin U$ or $v \notin U$. However, $C \setminus \{uv\}$ is a sequence of edges $w_1w_2, \dots, w_{k-1}w_k$ such that $u = w_1, v = w_k$ and $w_iw_{i+1} \notin \phi_G(\Pi)$ for all $1 \le i \le k-1$. Consequently, since Π is a partition of V, there exists some $U \in \Pi$ such that

$$w_1 \in U \wedge w_2 \in U \wedge \ldots \wedge w_{k-1} \in U \wedge w_k \in U.$$

This contradicts $w_1 = u \notin U$ or $w_k = v \notin U$.

To show injectivity, let $\Pi = \{U_1, \dots, U_k\}$, $\Pi' = \{U'_1, \dots, U'_\ell\}$ be two decompositions of G. Suppose $\Pi \neq \Pi'$. Then (w.l.o.g.) there exist some $u, v \in V$ with $uv \in E$ and some $U_i \in \Pi$ such that $u, v \in U_i$ and for all $U'_j \in \Pi'$ it holds that $u \notin U'_j$ or $v \notin U'_j$. Thus, $uv \in \phi_G(\Pi')$ but $uv \notin \phi_G(\Pi)$, which means $\phi_G(\Pi) \neq \phi_G(\Pi')$.

For surjectivity, take some multicut $M \subseteq E$ of G. Let $\Pi = \{U_1, \dots, U_k\}$ collect the node sets of the connected components of the graph $(V, E \setminus M)$. Apparently, Π defines a decomposition of G. Moreover, it holds that $\phi_G(\Pi) = M$ by virtue of (5.1). □

For any multicut M of $G = (V, E)$, the characteristic vector $\mathbb{1}_M$ makes explicit for every pair $uv \in E$, whether u and v are in distinct components. To make explicit also for non-neighboring nodes, specifically, for all $uv \in F$ where $F \subset \binom{V}{2}$ and $F \cap E = \emptyset$, whether u and v are in distinct components, we define a lifting of the multicuts of G to multicuts of the augmented graph $\widehat{G} = (V, E \cup F)$.

Definition 5.2. Let $\widehat{G} = (V, E \cup F)$ be a graph obtained from G by adding the set of edges F. The composed map $\phi_{\widehat{G}} \circ \phi_G^{-1}$ is called the *lifting* of multicuts from G to $\widehat{G}$. For any multicut M of G, the set $\phi_{\widehat{G}} \circ \phi_G^{-1}(M)$ is called a multicut of $\widehat{G}$ *lifted from* G.

Definition 5.3. Let $G = (V, E)$ be a graph and $\widehat{G} = (V, E \cup F)$ an augmented graph. The convex hull of characteristic vectors of multicuts of G lifted to $\widehat{G}$ is called *lifted multicut polytope* w.r.t. G and $\widehat{G}$ and is denoted by

$$\mathsf{LMC}(G, \widehat{G}) = \operatorname{conv}\left\{\mathbb{1}_M \mid M \text{ multicut of } \widehat{G} \text{ lifted from } G\right\}. \tag{5.2}$$

We write $\mathsf{LMC} = \mathsf{LMC}(G, \widehat{G})$ for short whenever this is clear from the context.

Analogous to the multicut problem, we define the lifted multicut problem, which introduces additional coefficients for non-neighboring pairs of nodes.

Definition 5.4 (Lifted Multicut Problem)**.** Let $\theta \in \mathbb{R}^{E \cup F}$ be a vector associated with the edges of the augmented graph $\widehat{G} = (V, E \cup F)$. The *lifted multicut problem* w.r.t. G, $\widehat{G}$ and θ is to find a minimum multicut of $\widehat{G}$ lifted from G w.r.t. θ. More precisely, it is written in the form

$$\min_{x \in \mathsf{LMC}} \sum_{e \in E \cup F} \theta_e x_e \tag{$\mathrm{P}_{\mathsf{LMC}}$}$$

as a binary optimization problem with a linear objective function.

If $F = \emptyset$, then problem ($\mathrm{P}_{\mathsf{LMC}}$) specializes to the multicut problem (P_{MC}). If $F \neq \emptyset$, then problem ($\mathrm{P}_{\mathsf{LMC}}$) differs from the multicut problem w.r.t. $\widehat{G}$ and θ, since in general it holds that $\mathsf{LMC}(G, \widehat{G}) \subset \mathsf{MC}(\widehat{G})$, cf. Figures 5.1 and 5.2. In the lifted multicut problem the assignment x_{uv} indicates that the nodes u and v are connected in G by a path of edges labeled 0. This property can be used to penalize (by $\theta_{uv} > 0$) or support (by $\theta_{uv} < 0$) those decompositions of G for which u and v are in distinct components. If u and v are not neighbors in G, then such costs are also called *non-local*.

5.3 LIFTED MULTICUT POLYTOPE

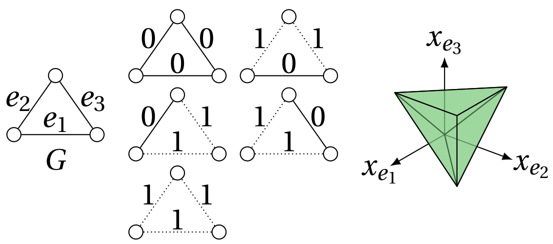

Figure 5.1: For any connected graph G (left), the characteristic vectors of multicuts of G (middle) span, as their convex hull in $\mathbb{R}^E$, the multicut polytope of G (right), a 01-polytope that is $|E|$-dimensional (Chopra and Rao, 1993).

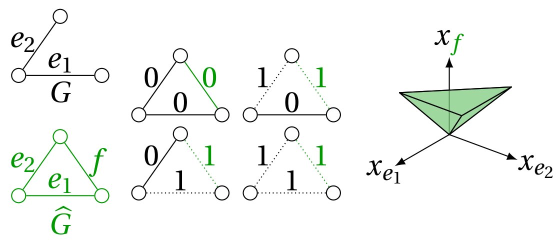

Figure 5.2: For any connected graph $G = (V, E)$ (top left) and any graph $\widehat{G} = (V, E \cup F)$ (bottom left), the characteristic vectors of multicuts of $\widehat{G}$ that are lifted from G (middle) span, as their convex hull in $\mathbb{R}^{E \cup F}$, the lifted multicut polytope w.r.t. G and $\widehat{G}$ (right), a 01-polytope that is $|E \cup F|$-dimensional (Thm. 5.1).

In this section, we study the geometry of the lifted multicut polytope LMC defined by (5.2). To this end, we first state a description of LMC in terms of linear inequalities and integer constraints. Recall from Section 1.2.3 the cycle inequality

$$x_f \leq \sum_{e \in C \setminus \{f\}} x_e \tag{5.3}$$

associated with a cycle C. Additionally, we introduce another two types of inequalities. For each $uv \in F$ and any uv-path P in G, we define the *path inequality*

$$x_{uv} \le \sum_{e \in P} x_e. \tag{5.4}$$

Furthermore, for each $uv \in F$ and any uv-cut $\delta(U)$ in G, we define the *cut inequality*

$$1 - x_{uv} \le \sum_{e \in \delta(U)} 1 - x_e. \tag{5.5}$$

Lemma 5.2. *A vector $x \in \{0,1\}^{E \cup F}$ is the characteristic vector of a multicut of $\widehat{G}$ lifted from G if, and only if, additional to the cycle inequalities in G, it satisfies all path and cut inequalities. In other words,*

$$\begin{aligned} \mathsf{LMC} = \operatorname{conv}\big\{x \in \{0,1\}^{E \cup F} \mid\ & x \textit{ satisfies } (5.3) \textit{ for all cycles } C \textit{ of } G, \\ & x \textit{ satisfies } (5.4) \textit{ for all } uv \in F \textit{ and all } uv\textit{-paths } P \textit{ in } G, \\ & x \textit{ satisfies } (5.5) \textit{ for all } uv \in F \textit{ and all } uv\textit{-cuts } \delta(U) \textit{ in } G\big\}. \end{aligned}$$

Proof. Let $x \in \{0,1\}^{E \cup F}$ be such that $x = \mathbb{1}_M$ for a multicut M of $\widehat{G}$ lifted from G. Every cycle in G is a cycle in $\widehat{G}$. Moreover, for any $uv = f \in F$ and any uv-path P in G, it holds that $P \cup \{f\}$ induces a cycle in $\widehat{G}$. Therefore, x satisfies all inequalities (5.3) and (5.4). Assume x violates some inequality of (5.5). Then there is an edge $uv \in F$ and some uv-cut $\delta(U)$ in G such that $x_{uv} = 0$ and for all $e \in \delta(U)$ we have $x_e = 1$. Let Π be the partition of V corresponding to M according to Lemma 5.1. Then there exists some $U \in \Pi$ with $u \in U$ and $v \in U$. However, for any $ww' = \delta(U)$ it holds that $w \notin U$ or $w' \notin U$. This means the induced subgraph $(U, E \cap \binom{U}{2})$ is not connected, as $\delta(U)$ is a uv-cut. Hence, Π is not a decomposition of G, which is a contradiction.

Now, suppose $x \in \{0,1\}^{E \cup F}$ satisfies the inequalities (5.3)–(5.5) as specified. We show first that $M = x^{-1}(1)$ is a multicut of $\widehat{G}$. Assume the contrary, then there is a cycle C in $\widehat{G}$ and some edge e such that $C \cap M = \{e\}$. For every $uv = f \in F \cap C \setminus \{e\}$ there exists a uv-path P in G such that $x_e = 0$ for all $e \in P$. Otherwise there would be some uv-cut in G violating (5.5), as G is connected. If we replace every such f with its associated path P in G, then the resulting cycle violates either (5.3) (if $e \in E$) or (5.4) (if $e \in F$). Thus, M is a multicut of $\widehat{G}$. By connectivity of G, the partition $\phi_{\widehat{G}}^{-1}(M)$ is a decomposition of both $\widehat{G}$ and G. Therefore, $(\phi_G \circ \phi_{\widehat{G}}^{-1})(M)$ is a multicut of G and hence M is indeed lifted from G. □

Note that, by Lemma 5.2, the system of all cycle inequalities for $\widehat{G}$ and all cut inequalities for G and $\widehat{G}$ is redundant as a description of LMC. Nonetheless, in a branch-and-cut algorithm it may be beneficial to separate certain inequalities for cycles of $\widehat{G}$ that are not included in (5.3) – (5.4), as we discuss in Section 5.4.

5.3.1 Dimension

In this section we show that LMC is full-dimensional as a polytope in $\mathbb{R}^{E \cup F}$. In order to prove this, we need to introduce some technical tools.

Definition 5.5. For any connected graph $G = (V, E)$ and $\widehat{G} = (V, E \cup F)$, the sequence $(F_n)_{n \in \mathbb{N}}$ of subsets of F defined below is called *hierarchy* of F w.r.t. G:

(a) $F_0 = \emptyset$

(b) For any $n \in \mathbb{N}$ and any $uv = f \in F$ it holds that:

$$uv \in F_n \iff \exists uv\text{-path } P \text{ in } G \text{ such that for all } w, w' \in V_P \text{ with } w \neq w', uv \neq ww' : \\ \text{either } ww' \notin F \text{ or } ww' \in F_j \text{ for some } j < n.$$

For $uv = f \in F$, any uv-path in G that satisfies the condition in (b) is called *f-hierarchical.*

The following lemma shows that the defined hierarchy is complete.

Lemma 5.3. *For any $f \in F$ there exists some $n \in \mathbb{N}$ such that $f \in F_n$.*

Proof. Let $uv = f \in F$ and let $d(u, v)$ denote the length of a shortest uv-path in G. Then $d(u, v) > 1$, because $F \cap E = \emptyset$. We proceed by induction on $d(u, v)$.

If $d(u, v) = 2$, there exists a $w \in V$ such that $uw \in E$ and $vw \in E$. Moreover, we have $uw \notin F$ and $vw \notin F$, as $F \cap E = \emptyset$. Thus $f \in F_1$.

If $d(u, v) = m$ with $m > 2$, consider any shortest uv-path P in G. Now, let $F' \subseteq F$ be such that, for any $ww' = f' \in F$, $f' \in F'$ iff $w \in P$ and $w' \in P$ and $f' \neq f$. If $F' = \emptyset$, then $f \in F_1$. Otherwise

$$\forall ww' \in F' : \quad d(w, w') < m \tag{5.6}$$

and thus

$$\forall f' \in F' \; \exists n_{f'} \in \mathbb{N} : \quad f' \in F_{n_{f'}} \tag{5.7}$$

by the induction hypothesis. Therefore, let $n = \max_{f' \in F'} n_{f'}$, then $f \in F_{n+1}$. □

Definition 5.6. The map $\ell \colon F \to \mathbb{N}$ defined by

$$\ell(f) = n \iff f \in F_n \text{ and } f \notin F_{n-1}$$

is called the *level function* of F.

Lemma 5.4. *For any $f \in F$ there exists a multicut M of $\widehat{G}$ lifted from G such that*

(a) $f \notin M$

(b) $\forall f' \in F \setminus \{f\}$ *with* $\ell(f') \geq \ell(f) : \quad f' \in M.$

Proof. For any $uv \in F$, let P be a uv-hierarchical path in G and define

$$F' = \{ww' \in F \mid w \in P \text{ and } w' \in P\}, \\ F'' = F \setminus F'.$$

Now, let M be the multicut of $\widehat{G}$ lifted from G that corresponds to the partion of G into P and otherwise all singleton components. It holds that $uv \notin M$ and $f' \notin M$ for all $f' \in F'$. Furthermore, $f' \in M$ for all $f' \in F''$. Since any $f' \in F \setminus \{f\}$ with $\ell(f') \geq \ell(f)$ satisfies $f' \in F''$, the claim follows. □

Definition 5.7. A 01-vector $x \in \{0,1\}^{E\cup F}$ is *feasible* if $x \in \mathrm{LMC}(G,\widehat{G})$. We further say that x is f*-feasible* for some $f \in F$ if the associated multicut satisfies the conditions of Lemma 5.4.

We are now ready to prove Theorem 5.1 below. To this end, we construct $|E\cup F|+1$ multicuts of $\widehat{G}$ lifted from G whose characteristic vectors are affine independent points. The strategy is to construct for any $e \in E\cup F$ a multicut that does not include e, but "as many other edges as possible". The level function indicates which edges can be included in the multicut.

Theorem 5.1 (Dimension)**.** *It holds that*

$$\dim \mathrm{LMC} = |E\cup F|.$$

Proof. The all-one vector $\mathbb{1} \in \{0,1\}^{E\cup F}$ is feasible. For any $e \in E$, the vector $x^e \in \{0,1\}^{E\cup F}$ such that $x^e_e = 0$ and $x^e_{E\setminus\{e\}} = 1$ and $x^e_F = 1$ is feasible. For any $f \in F$, let $x^f \in \{0,1\}^{E\cup F}$ denote an f-feasible vector such that some f-hierarchical path is the only non-singleton component of the associated partition.

Now, for any $e \in E$, let $y^e = \mathbb{1} - x^e$. For any $f \in F_1$, define the vector $y^f \in \mathbb{R}^{E\cup F}$ via

$$y^f = \mathbb{1} - x^f - \sum_{\{e\in E \mid x^f_e = 0\}} y^e.$$

For any $n > 1$ and any $f \in F_n$, define the vector $y^f \in \mathbb{R}^{E\cup F}$ via

$$y^f = \mathbb{1} - x^f - \sum_{\{f'\in F \mid f'\neq f \text{ and } x^f_{f'}=0\}} y^{f'} - \sum_{\{e\in E \mid x^f_e = 0\}} y^e.$$

Here, $\ell(f') < \ell(f) \leq n$, by definition of f-feasibility. Thus, all $y^{f'}$ are well-defined by induction (over n). Observe that $\{y^e \mid e \in E\cup F\}$ is the unit basis in $\mathbb{R}^{E\cup F}$. Moreover, each of its elements is a linear combination of $\{\mathbb{1} - x^e \mid e \in E\cup F\}$, which is therefore linearly independent. Thus, $\{\mathbb{1}\} \cup \{x^e \mid e \in E\cup F\}$ is affine independent. Since all vectors in this set are feasible it follows that $\dim \mathrm{LMC} = |E\cup F|$. □

5.3.2 Conditions for Facets from Box Inequalities

We characterize those edges $e \in E\cup F$ for which the inequality $x_e \leq 1$ defines a facet of the lifted multicut polytope LMC.

Theorem 5.2. *The inequality $x_e \leq 1$ defines a facet of* LMC *if, and only if, there is no $uv \in F$ such that e connects a pair of uv-cut-vertices*[1].

Proof. Let $S = \{x \in \mathrm{LMC} \cap \mathbb{Z}^{E\cup F} \mid x_e = 1\}$ and put $\Sigma = \operatorname{conv} S$.

To show necessity, suppose there is some $uv = f \in F$ such that e connects a pair of uv-cut-vertices. Then, for any uv-path P in G, either $e \in P$ or e is a chord of P. We claim that we have $x_f = 1$ for any $x \in S$. This gives $\dim\Sigma \leq |E\cup F| - 2$, so the inequality $x_e \leq 1$

[1] For any $u, v \in V$, a uv*-cut-vertex* is a node $w \in V$ that lies on every uv-path of G.

cannot define a facet of LMC. If there are no uv-paths that have e as a chord, then $\{e\}$ is a uv-cut and the claim follows from the corresponding inequality of (5.5). Otherwise, every uv-path P that has e as a chord contains a subpath P' such that $P' \cup \{e\}$ induces a cycle. Thus, for any $x \in S$, the inequalities (5.3) or (5.4) (for $e \in E$ or $e \in F$, respectively) imply the existence of some $e_{P'} \in P'$ such that $x_{P'} = 1$. Let $\mathscr{P}$ denote the set of all such paths P'. Apparently, the collection $\bigcup_{P' \in \mathscr{P}} \{e_{P'}\} \cup \{e\}$ is a uv-cut set. This gives $x_f = 1$ via the inequality of (5.5) corresponding to this cut.

We turn to the proof of sufficiency. Assume there is no $uv = f \in F$ such that e connects a pair of uv-cut-vertices in G. The construction of an affine independent $|E \cup F|$-element set of feasible 01-vectors in S is analogous to the proof of Theorem 5.1, when the definition of the hierarchy is altered so as to consider only paths that do not have the edge e as a chord. Then, the assumption guarantees for any $uv = f \in F$ with $f \neq e$ the existence of an f-hierarchial path and thus an f-feasible $x \in S$ (where f-hierarchical and f-feasible are w.r.t. to the altered hierarchy). Hence, $\dim \Sigma = |E \cup F| - 1$, which means Σ is a facet of LMC. □

Next, we give conditions that contribute to identifying those edges $e \in E \cup F$ for which the inequality $0 \leq x_e$ defines a facet of the lifted multicut polytope LMC.

Theorem 5.3. *Let $e \in E \cup F$. In case $e \in E$, the inequality $0 \leq x_e$ defines a facet of* LMC *if, and only if, there is no triangle in $\widehat{G}$ that contains e. In case $uv = e \in F$, the inequality $0 \leq x_e$ defines a facet of* LMC *only if the following necessary conditions hold:*

(i) There is no triangle in $\widehat{G}$ that contains e.

(ii) The distance of any pair of uv-cut-vertices except uv itself is at least 3 in $\widehat{G}$.

(iii) There is no triangle in $\widehat{G}$ consisting of nodes s, s', t such that ss' is a uv-separating node set and t is a uv-cut-vertex.

Proof. Let $S = \{x \in \text{LMC} \cap \mathbb{Z}^{E \cup F} \mid x_e = 0\}$ and put $\Sigma = \operatorname{conv} S$.

Consider the case that $e \in E$. Let $G_{[e]}$ and $\widehat{G}_{[e]}$ be the graphs obtained from G and $\widehat{G}$, respectively, by contracting the edge e (and subsequently merging parallel edges). The lifted multicuts $x^{-1}(1)$ for $x \in S$ correspond bijectively to the multicuts of $\widehat{G}_{[e]}$ lifted from $G_{[e]}$. This implies $\dim \Sigma = \dim \text{LMC}(G_{[e]}, \widehat{G}_{[e]})$. The claim follows from Theorem 5.1 and the fact that $\widehat{G}_{[e]}$ has $|E \cup F| - 1$ many edges if and only if e is not contained in any triangle in $\widehat{G}$.

Now, suppose $uv = e \in F$. We show necessity of the conditions (i)-(iii) by proving that if any of them is violated, then all $x \in S$ satisfy some additional equation and thus, $\dim \Sigma \leq |E \cup F| - 2$.

First, assume that (i) is violated. Hence, there are edges $e', e'' \in E'$ such that $\{e, e', e''\}$ induces a triangle in $\widehat{G}$. Every $x \in S$ satisfies the cycle inequalities

$$x_{e'} \leq x_e + x_{e''}, \tag{5.8}$$

$$x_{e''} \leq x_e + x_{e'}. \tag{5.9}$$

Thus, by (5.8), (5.9) and $x_e = 0$, every $x \in S$ satisfies $x_{e'} = x_{e''}$.

Next, assume that (ii) is violated. Consider a violating pair $u'v' \neq uv, u' \neq v'$ of uv-cut-vertices. For every $x \in S$, there exists a uv-path P in G with $x_P = 0$, as $x_e = 0$. Any such path P has a sub-path P' from u' to v' because u' and v' are uv-cut-vertices. We distinguish the following cases.

- If the distance of u' and v' in $\widehat{G}$ is 1, then $u'v' \in E \cup F$. If $u'v' \in P$, then $x_{u'v'} = 0$ because $x_P = 0$. Otherwise, $x_{u'v'} = 0$ by $x_{P'} = 0$ and the cycle/path inequality

$$x_{u'v'} \leq \sum_{f \in P'} x_f .$$

  Thus $x_{u'v'} = 0$ for all $x \in S$.

- If the distance of u' and v' in $\widehat{G}$ is 2, there is a $u'v'$-path in $\widehat{G}$ consisting of two distinct edges $e', e'' \in E \cup F$. We show that all $x \in S$ satisfy $x_{e'} = x_{e''}$:
  - If $e' \in P$ and $e'' \in P$, then $x_{e'} = x_{e''} = 0$ because $x_P = 0$.
  - If $e' \in P$ and $e'' \notin P$ then $x_{e'} = x_{e''} = 0$ by $x_P = 0$ and the cycle/path inequality

$$x_{e''} \leq \sum_{f \in P' \setminus \{e'\}} x_f .$$

  - If $e' \notin P$ and $e'' \notin P$ then $x_{e'} = x_{e''}$ by $x_P = 0$ and the cycle/path inequalities

$$x_{e''} \leq x_{e'} + \sum_{f \in P'} x_f ,$$
$$x_{e'} \leq x_{e''} + \sum_{f \in P'} x_f .$$

Finally, assume that (iii) is violated. Hence, there exists a uv-cut-vertex t and a uv-separating set of vertices $\{s, s'\}$ such that $\{ts, ts', ss'\}$ induces a triangle in $\widehat{G}$. We have that all $x \in S$ satisfy $x_{ss'} = x_{ts} + x_{ts'}$ as follows. At most one of x_{ts} and $x_{ts'}$ is 1, because t is a uv-cut-vertex and $\{s, s'\}$ is uv-separating as well. Moreover, it holds that $x_{ts} + x_{ts'} = 0$ if and only if $x_{ss'} = 0$ by the associated triangle inequality. □

5.3.3 Characterization of Facets from Cycle and Path Inequalities

Next, we characterize those inequalities of (5.3) for cycles in G and (5.4) that are facet-defining for LMC. Chopra and Rao (1993) have shown that a cycle inequality defines a facet of the multicut polytope MC if, and only if, the associated cycle is chordless. We establish a similar characterization of those cycle and path inequalities in the minimal description of LMC that are facet-defining. For clarity, we introduce some notation: For any cycle C of G and any $f \in C$, let

$$S(f, C) = \left\{ x \in \mathrm{LMC} \cap Z^{E \cup F} \mid x_f = \sum_{e \in C \setminus \{f\}} x_e \right\},$$
$$\Sigma(f, C) = \mathrm{conv}\, S(f, C).$$

Similarly, for any $uv = f \in F$ and any uv-path P in G, let

$$\begin{aligned} S(f,P) &= \Big\{x \in \mathrm{LMC} \cap \mathbb{Z}^{E\cup F} \mid x_{uv} = \sum_{e\in P} x_e\Big\}, \\ \Sigma(f,P) &= \operatorname{conv} S(f,P). \end{aligned}$$

Theorem 5.4. *The following assertions hold true:*

(a) For any cycle C in G and any $f \in C$, the polytope $\Sigma(f,C)$ is a facet of LMC *if, and only if, C is chordless in $\widehat{G}$.*

(b) For any edge $uv = f \in F$ and any uv-path P in G, the polytope $\Sigma(f,P)$ is a facet of LMC *if, and only if, $P \cup \{f\}$ induces a chordless cycle in $\widehat{G}$.*

Proof. First we show that for any chordal cycle $\widehat{C}$ in $\widehat{G}$ and any $f \in \widehat{C}$, the associated cycle inequality

$$x_f \le \sum_{e\in \widehat{C}\setminus\{f\}} x_e \tag{5.10}$$

is not facet-defining for $\mathrm{MC}(\widehat{G})$. This implies that (5.10) is not facet-defining for $\mathrm{LMC}(G,\widehat{G})$ as $\mathrm{LMC}(G,\widehat{G}) \subseteq \mathrm{MC}(\widehat{G})$ and $\dim \mathrm{LMC}(G,\widehat{G}) = \dim \mathrm{MC}(\widehat{G})$. Hence, this shows necessity for both (a) and (b).

For this purpose, consider some cycle $\widehat{C}$ of $\widehat{G}$ with a chord $uv = f' \in E \cup F$. We may write $\widehat{C} = P_1 \cup P_2$ where P_1 and P_2 are edge-disjoint uv-paths such that $C_1 = P_1 \cup \{f'\}$ and $C_2 = P_2 \cup \{f'\}$ are cycles in $\widehat{G}$. Let $f \in \widehat{C}$, then either $f \in P_1$ or $f \in P_2$. W.l.o.g. we may assume $f \in P_1$. The inequalities

$$x_f \le \sum_{e\in C_1\setminus\{f\}} x_e \tag{5.11}$$

$$x_{f'} \le \sum_{e\in C_2\setminus\{f'\}} x_e \tag{5.12}$$

are both valid for $\mathrm{MC}(\widehat{G})$. Moreover, since $f' \in C_1$, (5.11) and (5.12) imply (5.10) via

$$\begin{aligned} x_f \le \sum_{e\in C_1\setminus\{f\}} x_e &= \sum_{e\in C_1\setminus\{f,f'\}} x_e + x_{f'} \\ &\le \sum_{e\in C_1\setminus\{f,f'\}} x_e + \sum_{e\in C_2\setminus\{f'\}} x_e \\ &= \sum_{e\in \widehat{C}\setminus\{f\}} x_e. \end{aligned}$$

Thus, (5.10) is not facet-defining for $\mathrm{MC}(\widehat{G})$.

For the proof of sufficiency, suppose the cycle C of G is chordless in $\widehat{G}$ and let $f \in C$. Let Σ be a facet of LMC such that $\Sigma(f,C) \subseteq \Sigma$ and suppose it is induced by the inequality

$$\sum_{e\in E\cup F} a_e x_e \le \alpha \tag{5.13}$$

with $a \in \mathbb{R}^{E \cup F}$ and $\alpha \in \mathbb{R}$, i.e., $\Sigma = \operatorname{conv} S$, where

$$S = \left\{ x \in \mathsf{LMC} \cap \mathbb{Z}^{E \cup F} \mid a^\top x = \alpha \right\}.$$

For convenience, we also define the linear space

$$L = \left\{ x \in \mathbb{R}^{E \cup F} \mid a^\top x = \alpha \right\}.$$

As $0 \in S(f, C) \subseteq S$, we have $\alpha = 0$. We show that (5.13) is a scalar multiple of the cycle inequality (5.3) and thus $\Sigma(f, C) = \Sigma$.

Let $y \in \{0, 1\}^{E \cup F}$ be defined by

$$y_C = 0, \quad y_{(E \cup F) \setminus C} = 1,$$

i.e. all edges except C are cut. Then $y \in S(f, C) \subseteq S$, since C is chordless. For any $e \in C \setminus \{f\}$, the vector $x \in \{0, 1\}^{E \cup F}$ with

$$x_{C \setminus \{f, e\}} = 0, \quad x_{(E \cup F) \setminus C \cup \{f, e\}} = 1$$

satisfies $x \in S(f, C) \subseteq S$. Therefore, $y - x \in L$ and thus

$$\forall e \in C \setminus \{f\}: \quad a_e = -a_f. \tag{5.14}$$

It remains to show that $a_e = 0$ for all edges $e \in E \cup F \setminus C$. We proceed by considering edges from E and F separately. We consider the nodes $u, v \in V$ such that $uv = e$. W.l.o.g., we assume that v does not belong to C (due to the fact that C does not have a chord in $\widehat{G}$).

First, suppose $e \in E$ and distinguish the following cases:

(i) If e connects two nodes not contained in C or it is the only edge connecting some node in C to v, then for $x \in \{0, 1\}^{E \cup F}$, defined by

$$x_C = 0, \quad x_e = 0, \quad x_{(E \cup F) \setminus (C \cup \{e\})} = 1,$$

it holds that $x \in S(f, C) \subseteq S$. Therefore, $y - x \in L$, which evaluates to $a_e = 0$.

(ii) Otherwise, let $E_{C,v} = \{u'v \in E \cup F \mid u' \text{ belongs to } C\}$ denote the set of edges in $E \cup F$ that connect v to some node in C. By assumption, we have that $|E_{C,v}| \geq 2$. Now, pick some direction on C and traverse C from one endpoint of f to the other endpoint of f. We may order the edges $E_{C,v} = \{e_1, \ldots, e_k\}$ such that the endpoint of e_i appears before the endpoint of e_{i+1} in the traversal of C. We show that $a_{e_i} = 0$ for all $1 \leq i \leq k$:

For the vector $x \in \{0, 1\}^{E \cup F}$ defined by

$$x_{e'} = \begin{cases} 0 & \text{if } e' \in C \\ 0 & \text{if } e' \in E_{C,v} \\ 1 & \text{else,} \end{cases}$$

it holds that $x \in S(f,C) \subseteq S$. Therefore, $y - x \in L$ and thus

$$\sum_{1 \le i \le k} a_{e_i} = 0. \tag{5.15}$$

Consider the $m \in \{1, \dots, k\}$ such that $e = e_m$. For any i with $1 \le i \le m-1$, consider the following construction that is also illustrated in Figure 5.3. Let $e' \in C$ be some edge between the endpoints of e_i and e_{i+1}. If $e_i \in E$, define $x \in \{0,1\}^{E \cup F}$ via

$$\begin{aligned} & x_f = x_{e'} = 1 \\ & x_{C \setminus \{f,e'\}} = 0 \\ \forall j \le i: \quad & x_{e_j} = 0 \\ \forall j > i: \quad & x_{e_j} = 1 \end{aligned}$$

If $e_i \in F$, define $x \in \{0,1\}^{E \cup F}$ via

$$\begin{aligned} & x_f = x_{e'} = 1 \\ & x_{C \setminus \{f,e'\}} = 0 \\ \forall j \le i: \quad & x_{e_j} = 1 \\ \forall j > i: \quad & x_{e_j} = 0 \end{aligned}$$

Either way, it holds that $x \in S(f,C) \subseteq S$ and thus $y - x \in L$. If $e_i \in E$, this yields

$$0 = \sum_{1 \le j \le i} a_{e_j} - a_f - a_{e'} = \sum_{1 \le j \le i} a_{e_j}$$

by (5.14). If $e_i \in F$, we similarly obtain

$$0 = \sum_{i+1 \le j \le k} a_{e_j} - a_f - a_{e'} = \sum_{i+1 \le j \le k} a_{e_j}.$$

and thus, together with (5.15), $\sum_{1 \le j \le i} a_{e_j} = 0$ as well. Applying this argument repeatedly from $i = 1$ to $i = m-1$, we conclude that $a_{e_1} = \dots = a_{e_{m-1}} = 0$. By reversing the order of the edges in $E_{C,v}$, it can be shown analogously that $a_{e_k} = a_{e_{k-1}} = \dots = a_{e_{m+1}} = 0$. Hence, by (5.15), $a_e = a_{e_m} = 0$. Note that, since $e_j \in E \cup F$, we have also shown that $a_{e'} = 0$ for every $e' \in F$ that connects some node in the cycle C to a neighbor (within G) of another node in C.

Next, suppose $e \in F$ and distinguish the following additional cases:

(iii) Suppose there is a uv-path P' in G that does not contain any node from C. Define $x \in \{0,1\}^{E \cup F}$ via

$$x_{e'} = \begin{cases} 0 & \text{if } e' \in C \\ 0 & \text{if } e' = e \\ 0 & \text{if } e' \in P' \text{ or } e' \text{ is a chord of } P' \\ 1 & \text{else.} \end{cases}$$

Figure 5.3: The figure illustrates the argument from case (ii) in the proof of Theorem 5.4 for the cycle $C = \{f, e', e''\}$. In this example, $e_3 = e$, $e_1 \in F$ and $e_2 \in E$. The left multicut is chosen for $i = 1$ and the right one for $i = 2$ (dotted edges are cut).

Then $x \in S(f, C) \subseteq S$ and thus $y - x \in L$. This gives

$$a_e + \sum_{e' \in P'} a_{e'} + \sum_{e' \text{ chord of } P'} a_{e'} = 0.$$

We argue that all terms except a_e vanish by induction over the length of the path P', which we denote by $d(P')$. If $d(P') = 2$, then P' does not have any chords from F, thus $a_e = 0$, because $a_{e'} = 0$ for all $e' \in E$ as shown previously in the cases (i) and (ii). If $d(P') > 2$, then for any chord $e' \in F$ of P' the corresponding subpath has shorter length. The induction hypothesis provides $a_{e'} = 0$ and hence we conclude $a_e = 0$.

(iv) Suppose that (w.l.o.g.) u is contained in C. Further, assume there is another node w in C and a vw-path P' in G such that P' and C are edge-disjoint. We argue inductively over the path length $d(P')$. If $d(P') = 1$, then P' consists of only one edge from E, which is e itself. Recall that this situation is in fact already covered by case (ii). If $d(P') > 1$, then we employ an argument similar to (ii) as follows. Let $F_{C,v} = \{u'v \in F \mid u' \text{ belongs to } C\} = \{f_1, \ldots, f_k\}$ be the set of edges in F that connect v to some node in C. We can (w.l.o.g.) ignore edges from E here, since then we could replace P' by a path of length 1. Again, assume the f_i are ordered such that the endpoint of f_i appears before the endpoint of f_{i+1} on C in a traversal from f to itself. For the vector $x \in \{0,1\}^{E \cup F}$ defined by

$$x_{e'} = \begin{cases} 0 & \text{if } e' \in C \\ 0 & \text{if } e' \in P' \\ 0 & \text{if } e' = u'v' \text{ is a chord of } P' \text{ and neither } u' \text{ nor } v' \text{ belongs to } C \\ 0 & \text{if } e' = u'v' \text{ where } v' \neq v \text{ belongs to } P' \text{ and } u' \text{ belongs to } C \\ 0 & \text{if } e' \in F_{C,v} \\ 1 & \text{else,} \end{cases}$$

it holds that $x \in S(f, C) \subseteq S$ and thus $y - x \in L$. By cases (i) – (iii) and the induction hypothesis, we know that $a_{e'} = 0$ for all $e' \in E \cup F$ that are covered by the first four lines in the definition of x. Therefore, the fact that $a^\top x = 0$ reduces to

$$\sum_{1 \leq i \leq k} a_{f_i} = \sum_{e' \in F_{C,v}} a_{e'} = 0. \tag{5.16}$$

Now, let m be the highest index such that the endpoint of f_m appears before w on C. For any i with $1 \leq i \leq m$, pick an edge $e' \in C$ between the endpoint of f_i and the endpoint of f_{i+1} and before w on C. Define $x \in \{0,1\}^{E \cup F}$ via

$$x_g = \begin{cases} 0 & \text{if } g \in C \setminus \{f, e'\} \\ 0 & \text{if } g \in P' \\ 0 & \text{if } g = u'v' \text{ is a chord of } P' \text{ and neither } u' \text{ nor } v' \text{ belongs to } C \\ 0 & \text{if } g = u'v', \text{ where } v' \neq v \text{ belongs to } P' \text{ and } u' \text{ appears after } w \text{ on } C \\ 0 & \text{if } g = f_j \ \forall j > i \\ 1 & \text{else.} \end{cases}$$

Then, it holds that $x \in S(f, C) \subseteq S$ and thus $y - x \in L$. This yields, after removing all zero terms once more,

$$\sum_{i+1 \leq j \leq k} a_{f_j} = 0. \tag{5.17}$$

Together with (5.16), we obtain

$$\sum_{1 \leq j \leq i} a_{f_i} = 0. \tag{5.18}$$

Applying this argument repeatedly for $i = 1$ to $i = m$, we conclude $a_{f_1} = \ldots = a_{f_m} = 0$. Similarly, we obtain $a_{f_k} = a_{f_{k-1}} = \ldots = a_{f_{m+2}} = 0$, by reversing the direction of traversal of C and employing the same reasoning. We can conclude $a_{f_{m+1}} = 0$ via (5.16), since it might be the case that $f_{m+1} = vw$. In particular, we showed that $a_e = 0$.

(v) Finally, consider an arbitrary uv-path P' in G (that is not covered by any previous case). We perform induction over the path length $d(P')$. If $d(P') = 2$, then the only remaining case is such that P' is edge-disjoint from C but shares a single vertex $w \not\subset uv$ with C. Define the vector $x \in \{0,1\}^{E \cup F}$ by

$$x_{e'} = \begin{cases} 0 & \text{if } e' \in C \\ 0 & \text{if } e' = e \\ 0 & \text{if } e' \in P' \\ 0 & \text{if } e' = u'v \text{ where } u' \text{ belongs to } C \\ 0 & \text{if } e' = uv' \text{ where } v' \text{ belongs to } C \\ 1 & \text{else.} \end{cases}$$

It holds that $x \in S(f, C) \subseteq S$ and thus $y - x \in L$, which implies $a_e = 0$ after removing all zero terms from $a^\top x = 0$ (use previous cases).

If $d(P') > 2$, define the vector $x \in \{0,1\}^{E\cup F}$ by

$$x_{e'} = \begin{cases} 0 & \text{if } e' \in C \\ 0 & \text{if } e' = e \\ 0 & \text{if } e' \in P' \text{ or } e' \text{ is a chord of } P' \\ 0 & \text{if } e' = u'v' \text{ where } u' \text{ belongs to } C \text{ and } v' \text{ belongs to } P' \\ 1 & \text{else.} \end{cases}$$

It holds that $x \in S(f,C) \subseteq S$ and thus $y - x \in L$. From the previous cases and the induction hypothesis we can conclude that all terms except a_e vanish from the equation $a^\top x = 0$ and hence $a_e = 0$.

The proof of sufficiency in the second assertion is completely analogous (replace C by $P \cup \{f\}$). The chosen multicuts remain valid, because f is the only edge in the cycle that is not contained in E. □

Any inequality w.r.t some cycle C in $\widehat{G}$ and $f \in C$ is valid for $\mathrm{LMC}(G,\widehat{G})$ as it is valid for $\mathrm{MC}(\widehat{G}) \supseteq \mathrm{LMC}(G,\widehat{G})$. For any cycle inequality to define a facet of $\mathrm{LMC}(G,\widehat{G})$ it is necessary that the associated cycle is chordless, as is shown in the proof of Theorem 5.4. In general, however, chordlessness is not a sufficient condition if the inequality is neither of the form (5.3) for a cycle C in G nor of the form (5.4). As a simple example, consider the graph $\widehat{G}$ depicted in Figure 5.2. Here, the cycle inequality $x_{e_1} \leq x_f + x_{e_2}$ is dominated by the cut inequality $x_{e_1} \leq x_f \iff 1 - x_f \leq 1 - x_{e_1}$.

5.3.4 Necessary Conditions for Facets from Cut Inequalities

Next, we consider the cut inequalities (5.5). Our goal is to constrain the class of cuts that give rise to facet-defining inequalities. Nontrivial examples of cuts whose associated inequalities fail to define facets of LMC are shown in Figure 5.4.

Obviously, for any $uv \in F$, we can restrict ourselves uv-cuts $\delta(U)$ that are *minimal*, i.e. any proper subset of $\delta(U)$ is not a uv-cut. Therefore, we assume (w.l.o.g.) that $U \subset V$ is such that $u \in U$ and $v \in V \setminus U$. For clarity, define

$$\begin{aligned} S(uv,U) &= \Big\{ x \in \mathrm{LMC} \cap Z^{E\cup F} \mid 1 - x_{uv} = \sum_{e\in\delta(U)} 1 - x_e \Big\}, \\ \Sigma(uv,U) &= \operatorname{conv} S(uv,U). \end{aligned}$$

Definition 5.8. Let $U \subseteq V$ such that $u \in U$, $v \notin U$ and $\delta(U)$ is a minimal uv-cut. An induced subgraph $H = (V_H, E_H)$ of G is called *properly* (uv,U)*-connected* if

$$u \in V_H, v \in V_H \text{ and } |E_H \cap \delta(U)| = 1.$$

It is called *improperly* (uv,U)*-connected* if

$$V_H \subseteq U \text{ or } V_H \subseteq V \setminus U.$$

It is called (uv,U)*-connected* if it is properly or improperly (uv,U)-connected.

Lemma 5.5. *For any $uv \in F$ and any $U \subseteq V$ with $u \in U$, $v \notin U$ and $\delta(U)$ a minimal uv-cut, the following holds:*

(a) Every $x \in S(uv, U)$ defines a decomposition of G into (uv, U)-connected components. That is, every maximal component of the graph $(V, \{e \in E \mid x_e = 0\})$ is (uv, U)-connected. At most one of these is properly (uv, U)-connected. It exists if, and only if, $x_{uv} = 0$.

(b) For every (uv, U)-connected component $H = (V_H, E_H)$ of G, the vector $x \in \{0,1\}^{E \cup F}$ defined by

$$x_{u'v'} = 0 \iff u' \in V_H \text{ and } v' \in V_H \qquad \forall u'v' \in E \cup F$$

satisfies $x \in S(uv, U)$.

Proof. (a) Take some $x \in S(uv, U)$. Let $E_0 = \{e \in E \mid x_e = 0\}$ and consider $G_0 = (V, E_0)$.

If $x_{uv} = 1$ then for all $e \in \delta(U)$ it holds that $x_e = 1$. Thus, every component of G_0 is improperly (uv, U)-connected.

If $x_{uv} = 0$ then there is some $e \in \delta(U)$ such that

$$x_e = 0 \text{ and } \forall e' \in \delta(U) \setminus \{e\} : x_{e'} = 1. \tag{5.19}$$

Let $H = (V_H, E_H)$ be the maximal component of G_0 with

$$e \in E_H. \tag{5.20}$$

Clearly,

$$\forall e' \in \delta(U) \setminus \{e\} : \ e' \notin E_H \tag{5.21}$$

by (5.19) and definition of G_0. There is no uv-cut $\delta(W)$ with $x_{\delta(W)} = 1$, because this would imply $x_{uv} = 1$. Thus, there exists a uv-path P in G with $x_P = 0$, as G is connected. Any such path P has $e \in P$, as $P \cap \delta(U) \neq \emptyset$ and $\delta(U) \cap E_0 = \{e\}$ and $P \subseteq E_0$. Thus,

$$u \in V_H \text{ and } v \in V_H \tag{5.22}$$

by (5.20). Therefore, $H = (V_H, E_H)$ is properly (vw, C)-connected, by (5.20), (5.21) and (5.22). Any other component of G_0 does not cross the cut $\delta(U)$, by (5.19), (5.20) and definition of G_0, and is thus improperly (uv, U)-connected.

(b) We have

$$\forall u'v' \in E : \quad x_{u'v'} = 0 \iff u'v' \in E_H \tag{5.23}$$

by the following argument:

- If $u'v' \in E_H$, then $u' \in V_H$ and $v' \in V_H$, as $H = (V_H, E_H)$ is a graph. Thus, $x_{u'v'} = 0$, by definition of x.
- If $u'v' \notin E_H$ then $u' \notin V_H$ or $v' \notin V_H$, as $H = (V_H, E_H)$ is an induced subgraph of G. Thus, $x_{u'v'} = 1$, by definition of x.

Consider the decomposition of G into $H = (V_H, E_H)$ and otherwise singleton components. The set $E_1 = \{e \in E \mid x_e = 1\}$ contains precisely those edges that straddle distinct components of this decomposition and therefore E_1 is a multicut of G. In particular, the vector x satisfies all cycle inequalities w.r.t. cycles in G.
For any $u'v' \in F$ and any $u'v'$-path P in G, distinguish two cases:

- If $P \subseteq E_H$, then $u' \in V_H$ and $v' \in V_H$, as $H = (V_H, E_H)$ is a graph. Thus, $x_{u'v'} = 0$ by definition of x. Moreover, $x_P = 0$ by (5.23). Hence, the corresponding path inequality (5.4) evaluates to $0 = 0$.

- Otherwise, there exists an $e \in P$ such that $e \notin E_H$. Therefore, $x_e = 1$ by (5.23). Thus, (5.4) holds again, as the right-hand side is at least 1.

For any $u'v' \in F$ and any uv-cut $\delta(W)$, distinguish two cases:

- If $\delta(W) \cap E_H = \emptyset$, then $V_H \subseteq W$ or $V_H \subseteq V \setminus W$, thus $u' \notin V_H$ or $v' \notin V_H$. Therefore, $x_{u'v'} = 1$ by definition of x. Moreover, $x_{\delta(W)} = 1$ by (5.23). Thus, the corresponding cut inequality (5.5) evaluates to $0 = 0$.

- Otherwise, there exists an $e \in \delta(W)$ such that $e \in E_H$. Therefore, $x_e = 0$ by (5.23). Thus, (5.5) holds again, as the right-hand side is at least 1.
  If $W = U$ and $u'v' = uv$, then $|\delta(U) \cap E_H| = 1$, since H is (in this case properly) (uv, U)-connected. Furthermore, $x_{uv} = 0$ by (5.23), which means (5.5) evaluates to $1 = 1$.

We can therefore conclude that $x \in S(uv, U)$. □

We denote by $\delta_{F\setminus\{uv\}}(U)$ the set of edges in F except uv that cross the cut, i.e.

$$\delta_{F\setminus\{uv\}}(U) = \left\{u'v' \in F \setminus \{uv\} \mid u' \in U, v' \notin U\right\}.$$

Let further

$$\widehat{G}(uv, U) = (V, \delta(U) \cup \delta_{F\setminus\{uv\}}(U))$$

denote the subgraph of $\widehat{G}$ that comprises all edges of the cut induced by U except uv. For any (uv, U)-connected subgraph $H = (V_H, E_H)$ of G, we denote by

$$F'_H = \{u'v' \in \delta_{F\setminus\{uv\}}(U) \mid u' \in V_H \text{ and } v' \in V_H\}$$

the set of those edges in $u'v' \in \delta_{F\setminus\{uv\}}(U)$ such that H is also $(u'v', U)$-connected.

Theorem 5.5. *For any $uv \in F$ and any $U \subseteq V$ with $u \in U$, $v \notin U$ and $\delta(U)$ a minimal uv-cut, the polytope $\Sigma(uv, U)$ is a facet of* LMC *only if the following necessary conditions hold:*

C1 For any $e \in \delta(U)$, there exists some (uv, U)-connected subgraph $H = (V_H, E_H)$ of G such that $e \in E_H$.

C2 For any $\emptyset \neq F' \subseteq \delta_{F\setminus\{uv\}}(U)$, there exists an edge $e \in \delta(U)$ and (uv, U)-connected subgraphs $H = (V_H, E_H)$ and $H' = (V_{H'}, E_{H'})$ of G such that

$$e \in E_H \text{ and } e \in E_{H'} \text{ and } |F' \cap F'_H| \neq |F' \cap F'_{H'}|.$$

C3 *For any* $f' \in \delta_{F\setminus\{uv\}}(U)$*, any* $\emptyset \neq F' \subseteq \delta_{F\setminus\{uv\}}(U) \setminus \{f'\}$ *and any* $k \in \mathbb{N}$*, there exist* (uv, U)*-connected subgraphs* $H = (V_H, E_H)$ *and* $H' = (V_{H'}, E_{H'})$ *with* $f' \in F'_H$ *and* $f' \notin F'_{H'}$ *such that*

$$|F' \cap F'_H| \neq k \text{ or } |F' \cap F'_{H'}| \neq 0.$$

C4 *For any* $u' \in U$*, any* $v' \in V \setminus U$ *and any* $u'v'$*-path* $P = (V_P, E_P)$ *in* $\widehat{G}(uv, U)$*, there exists a properly* (uv, U)*-connected subgraph* $H = (V_H, E_H)$ *of* G *such that*

$$\begin{aligned} & (u' \notin V_H \text{ or } \exists v'' \in V_P \cap V \setminus U : v'' \notin V_H) \\ \text{and} \quad & (v' \notin V_H \text{ or } \exists u'' \in V_P \cap U : u'' \notin V_H). \end{aligned}$$

C5 *For any cycle* $C = (V_C, E_C)$ *in* $\widehat{G}(uv, U)$*, there exists a properly* (uv, U)*-connected subgraph* $H = (V_H, E_H)$ *of* G *such that*

$$\begin{aligned} & (\exists u' \in V_C \cap U : u' \notin V_H) \\ \text{and} \quad & (\exists v' \in V_C \cap V \setminus U : v' \notin V_H). \end{aligned}$$

Proof. We show that if any of the conditions C1 – C5 is violated, then all $x \in S(uv, U)$ satisfy some additional equation and thus $\dim \Sigma(uv, U) \leq |E \cup F| - 2$, which implies that $\Sigma(uv, U)$ cannot be a facet of LMC by Theorem 5.1.

Assume that condition C1 does not hold (e.g. as in Figure 5.4a). Then there exists an $e \in \delta(U)$ such that no (uv, U)-connected subgraph of G contains e. Thus, for all $x \in S(uv, U)$ it holds that

$$x_e = 1 \tag{5.24}$$

by Lemma 5.5.

Assume that condition C2 does not hold. Then for any $e \in \delta(U)$ there exists some number $m \in \mathbb{N}$ such that for all (uv, U)-connected subgraphs $H = (V_H, E_H)$ with $e \in E_H$ it holds that $|F' \cap F'_H| = m$. Thus, we can write

$$\delta(U) = \bigcup_{m=0}^{|F'|} \delta^m(U),$$

where

$$\delta^m(U) = \left\{ e \in \delta(U) \mid m = |F' \cap F'_H| \ \forall (uv, U)\text{-connected } (V_H, E_H) \text{ with } e \in E_H \right\}.$$

It follows that for all $x \in S(uv, U)$ we have the equality

$$\sum_{m=0}^{|F'|} m \sum_{e \in \delta^m(U)} (1 - x_e) = \sum_{f' \in F'} (1 - x_{f'}) \tag{5.25}$$

by the following argument:

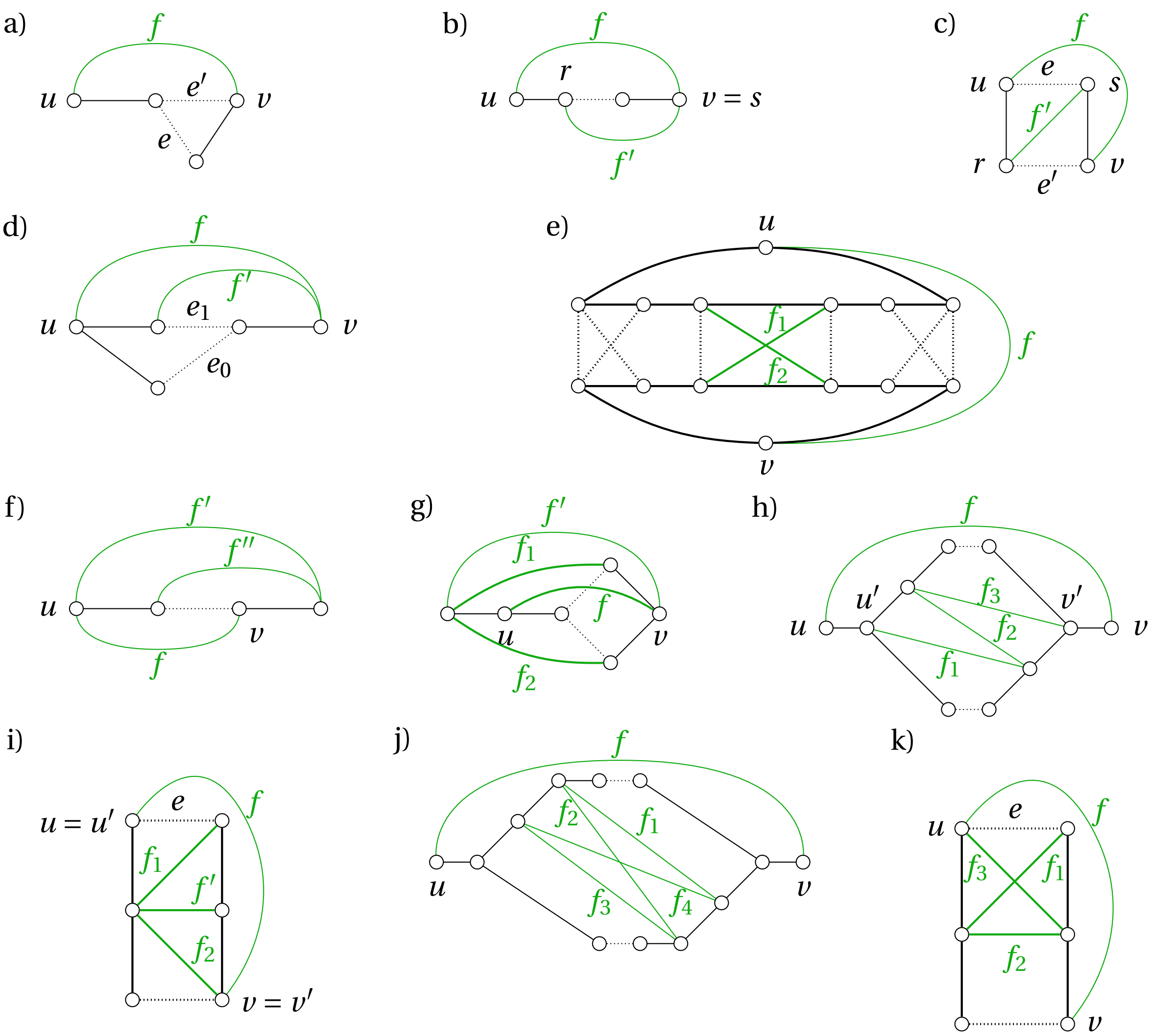

Figure 5.4: Depicted above are graphs $G = (V, E)$ (in black) and $\widehat{G} = (V, E \cup F)$ (F in green), distinct nodes $u, v \in V$ and a minimal uv-cut $\delta(U)$ in G (as dotted lines). In any of the above examples, one condition of Theorem 5.5 is violated and thus, $\Sigma(uv, U)$ is not a facet of the lifted multicut polytope LMC. **a)** Condition C1 is violated for e. **b)** Condition C2 is violated as r and s are connected in any (uv, U)-connected component. **c)** Condition C2 is violated as r and s are not connected in any (uv, U)-connected component. **d)** Condition C2 is violated. Specifically, $\delta^0(U) = \{e_0\}$ and $\delta^1(U) = \{e_1\}$ in the proof of Theorem 5.5. **e)** Condition C2 is violated for $F' = \{f_1, f_2\}$. **f)** Condition C3 is violated. **g)** Condition C3 is violated for $F' = \{f_1, f_2\}$ and $k = 1$. **h)** Condition C4 is violated for the $u'v'$-path $\{f_1, f_2, f_3\}$. **i)** Condition C4 is violated for the $u'v'$-path $\{e, f_1, f_2\}$. **j)** Condition C5 is violated for the cycle $\{f_1, f_2, f_3, f_4\}$. **k)** Condition C5 is violated for the cycle $\{e, f_1, f_2, f_3\}$.

- If $x_e = 1$ for all $e \in \delta(U)$, then $x_{f'} = 1$ for all $u'v' = f' \in F'$, since $\delta(U)$ is also a $u'v'$-cut. Thus, (5.25) evaluates to $0 = 0$.

- Otherwise there exists precisely one edge $e \in \delta(U)$ such that $x_e = 0$. Let m be such

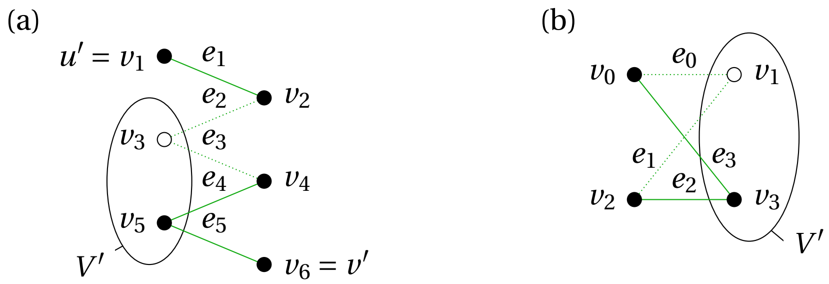

Figure 5.5: Depicted are the nodes (in black) and edges (in green) on a path **(a)** and on a cycle **(b)**, respectively. Nodes in the set V' are are either in V_H (filled circle) or not in V_H (empty circle). Consequently, pairs of consecutive edges are either cut (dotted lines) or not cut (solid lines).

that $e \in \delta^m(U)$. By definition of $\delta^m(U)$, there are exactly m edges $f' \in F'$ with $x_{f'} = 0$. Thus, (5.25) evaluates to $m = m$.

Assume that condition C3 does not hold. Then there exists an $f' \in \delta_{F\setminus\{uv\}}(U)$, a set $\emptyset \neq F' \subseteq \delta_{F\setminus\{uv\}}(U) \setminus \{f'\}$ and some $k \in \mathbb{N}$ such that for all (uv, U)-connected subgraphs $H = (V_H, E_H)$ and $H' = (V_{H'}, E_{H'})$ with $f' \in F'_H$ and $f' \notin F'_{H'}$ it holds that

$$|F' \cap F'_H| = k \text{ and } |F' \cap F'_{H'}| = 0.$$

In other words, for all $x \in S(uv, U)$ it holds that $x_{f'} = 0$ iff there are exactly k edges $f'' \in F'$ such that $x_{f''} = 0$. Similarly, it holds that $x_{f'} = 1$ iff for all $f'' \in F'$ we have $x_{f''} = 1$. Therefore, all $x \in S(uv, U)$ satisfy the additional equation

$$k(1 - x_{f'}) = \sum_{f'' \in F'} 1 - x_{f''}.$$

Assume that condition C4 does not hold. Then there exist $u' \in U$ and $v' \in V \setminus U$ and a $u'v'$-path $P = (V_P, E_P)$ in $\widehat{G}(uv, U)$ such that for every properly (uv, U)-connected subgraph $H = (V_H, E_H)$ of G it holds that

$$(u' \in V_H \text{ and } V_P \cap V \setminus U \subseteq V_H) \tag{5.26}$$

$$\text{or} \quad (v' \in V_H \text{ and } V_P \cap U \subseteq V_H). \tag{5.27}$$

Let $v_1 < \ldots < v_{|V_P|}$ be the linear order of the nodes V_P and let $e_1 < \ldots < e_{|E_P|}$ be the linear order of the edges E_P in the $u'v'$-path P. Now, all $x \in S(uv, U)$ satisfy the equation

$$x_{uv} = \sum_{j=1}^{|E_P|} (-1)^{j+1} x_{e_j} \tag{5.28}$$

by the following argument. It holds that $|E_P|$ is odd, as the path P alternates between the set U where it begins and $V \setminus U$ where it ends. Thus, we can write

$$\sum_{j=1}^{|E_P|} (-1)^{j+1} x_{e_j} = x_{e_1} - \sum_{j=1}^{(|E_P|-1)/2} (x_{e_{2j}} - x_{e_{2j+1}}). \tag{5.29}$$

Distinguish two cases:

- If $x_{uv} = 1$, then $x_{E_P} = 1$, by the cut inequalities (5.5) w.r.t. $\delta(U)$. Therefore, (5.29) and thus (5.28) evaluates to $1 = 1$.

- If $x_{uv} = 0$, then the decomposition of G defined by x contains precisely one properly (uv, U)-connected component $H = (V_H, E_H)$ of G, by Lemma 5.5. W.l.o.g. we may assume that (5.26) holds (otherwise exchange u and v). Consider the nodes V_P (as depicted in Figure 5.5a). It holds that $v_1 = u' \in V_H$, by (5.26). For every even j, $v_j \in V \setminus U$, by definition of P. Thus,

$$\forall \in \{1, \dots, (|E_P| - 1)/2\}: \quad v_{2j} \in V_H \tag{5.30}$$

by (5.26). Now consider the edges E_P (as depicted in Figure 5.5a). It holds that $e_1 = v_1 v_2 \in E_H$, as $v_1 \in V_H$ and $v_2 \in V_H$ and as $H = (V_H, E_H)$ is an induced subgraph of G. Thus,

$$x_{e_1} = 0 \tag{5.31}$$

by Lemma 5.5. For every $j \in \{1, \dots, (|E_P| - 1)/2\}$, distinguish two cases:

  – If $v_{2j+1} \in V_H$, then $e_{2j} = v_{2j} v_{2j+1} \in E_H$ and $e_{2j+1} = v_{2j+1} v_{2j+2} \in E_H$, because $v_{2j} \in V_H$ and $v_{2j+2} \in V_H$, by (5.30), and because $H = (V_H, E_H)$ is an induced subgraph of G. This implies

$$x_{e_{2j}} = 0 = x_{e_{2j+1}}. \tag{5.32}$$

  – If $v_{2j+1} \notin V_H$, then $e_{2j} = v_{2j} v_{2j+1}$ and $e_{2j+1} = v_{2j+1} v_{2j+2}$ straddle distinct components of the decomposition of G defined by x, because $v_{2j} \in V_H$ and $v_{2j+2} \in V_H$, by (5.30). This implies

$$x_{e_{2j}} = 1 = x_{e_{2j+1}}. \tag{5.33}$$

In any case, we have

$$\forall j \in \{1, \dots, (|E_P| - 1)/2\}: \quad x_{e_{2j}} - x_{e_{2j+1}} = 0. \tag{5.34}$$

Thus, (5.28) evaluates to $0 = 0$, by (5.29), (5.31) and (5.34).

Assume that condition C5 does not hold. Then, there exists a cycle $C = (V_C, E_C)$ in $\widehat{G}(uv, U)$ such that every properly (uv, U)-connected subgraph $H = (V_H, E_H)$ of G satisfies

$$V_C \cap U \subseteq V_H \tag{5.35}$$

$$\text{and} \quad V_C \cap V \setminus U \subseteq V_H. \tag{5.36}$$

Let $v_0 < \dots < v_{|V_C|-1}$ be an order on V_C such that $v_0 \in U$ and for all $j \in \{0, \dots, |E_C| - 1\}$ it holds that

$$e_j = v_j v_{j+1 \bmod |E_C|} \in E_C. \tag{5.37}$$

Now, all $x \in S(uv, U)$ satisfy the equation

$$\sum_{j=0}^{|E_C|-1} (-1)^j x_{e_j} = 0 \tag{5.38}$$

by the following argument. It holds that $|E_C|$ is even, as the cycle C alternates between the sets U and $V \setminus U$. Thus,

$$\sum_{j=0}^{|E_C|-1} (-1)^j x_{e_j} = \sum_{j=0}^{(|E_C|-2)/2} (x_{e_{2j}} - x_{e_{2j+1}}). \tag{5.39}$$

Distinguish two cases:

- If $x_{uv} = 1$, then $x_{E_C} = 1$, by the cut inequalities (5.5) w.r.t. $\delta(U)$. Therefore, (5.39) and thus (5.38) evaluates to $0 = 0$.

- If $x_{uv} = 0$, then the decomposition of G defined by x contains precisely one properly (uv, U)-connected component $H = (V_H, E_H)$ of G, by Lemma 5.5. W.l.o.g. we may assume that (5.35) holds (otherwise exchange u and v). Consider the nodes V_C (as depicted in Figure 5.5b). For every even j, we have that $v_j \in U$, by definition of C and the order. Thus,

$$\forall j \in \{0, \dots, (|E_C| - 2)/2\} : \quad v_{2j} \in V_H \tag{5.40}$$

by (5.35). Now, consider the edges E_C (as depicted in Fig. 5.5b). For every $j \in \{0, \dots, (|E_C| - 2)/2\}$, distinguish two cases:

  - If $v_{2j+1} \in V_H$, then $e_{2j} = v_{2j}v_{2j+1} \in E_H$ and $e_{2j+1} = v_{2j+1}v_{2j+2 \bmod |E_C|} \in E_H$, because $v_{2j} \in V_H$ and $v_{2j+2 \bmod |E_C|} \in V_H$, by (5.40), and because $H = (V_H, E_H)$ is an induced subgraph of G. This implies

$$x_{e_{2j}} = 0 = x_{e_{2j+1}}. \tag{5.41}$$

  - If $v_{2j+1} \notin V_H$, then $e_{2j} = v_{2j}v_{2j+1}$ and $e_{2j+1} = v_{2j+1}v_{2j+2 \bmod |E_C|}$ straddle distinct components of the decomposition of G defined by x, because $v_{2j} \in V_H$ and $v_{2j+2 \bmod |E_C|} \in V_H$, by (5.40). This implies

$$x_{e_{2j}} = 1 = x_{e_{2j+1}}. \tag{5.42}$$

  In any case, we have

$$\forall j \in \{0, \dots, (|E_C| - 2)/2\} : \quad x_{e_{2j}} - x_{e_{2j+1}} = 0. \tag{5.43}$$

  Thus, (5.38) evaluates to $0 = 0$, by (5.39) and (5.43). □

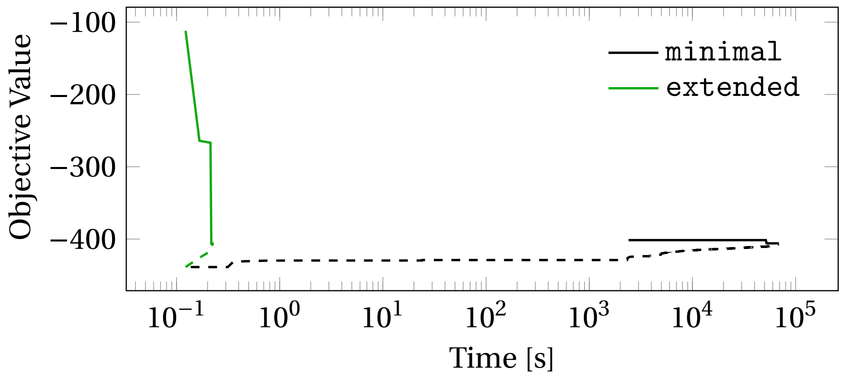

Figure 5.6: Compared above are the separation procedures `minimal` (black) and `extended` (green) in a branch-and-cut algorithm to solve an instance of the lifted multicut problem from Keuper et al. (2015b). Upper and lower bounds are depicted as solid and dashed lines, respectively.

5.4 ALGORITHMS

To study the practical relevance of geometric properties of LMC, we compare two separation procedures, `minimal` and `extended`, for lifted multicut polytopes. We implement these for the branch-and-cut algorithm in Gurobi. The procedure `minimal` is canonical and serves as a reference. It separates infeasible points by cycle inequalities for cycles in G as well as path and cut inequalities. Violated cycle and path inequalities are found by searching for shortest chordless paths. Violated cut inequalities are found by searching for minimum uv-cuts. The procedure `extended` is less canonical: It separates infeasible points by some cycle inequalities for cycles in $\widehat{G}$ and by cut inequalities. Violated cycle inequalities of $\widehat{G}$ are found by first seaching for paths and cycles as before but then replacing sub-paths by chords in $\widehat{G}$. Violated cut-inequalities are found as before but added to the problem only conditionally: For each violated cut inequality w.r.t. $uv \in F$ and $\delta(U)$ that we find, we search for a uv-path P in $\widehat{G}$ such that one of the cycle inequalities for the cycle formed by $P \cup \{uv\}$ is violated. If it exists, only the cycle inequality is added. Otherwise, the cut-inequality is added. The advantage of `extended` over `minimal` can be seen in Figure 5.6 for an instance of the lifted multicut problem from Keuper et al. (2015b) with $|V| = 126$, $|E| = 229$ and $|E \cup F| = 1860$.

5.5 CONCLUSION

We have defined a lifting of multicuts in a graph G to an augmented graph $\widehat{G}$, which makes connectedness in the associated partition explicit for non-neighboring pairs of nodes. Moreover, we have studied the lifted multicut polytope LMC that is associated with multicuts of G lifted to $\widehat{G}$. We established some conditions on the facet-defining property of the fundamental inequalities associated with lifted multicuts (box, cycle, path and cut inequalities). In particular, we characterized which cycles and paths in G give rise to facet-defining inequalities, thereby generalizing a result due to Chopra and Rao (1993). In the case of cut inequalities we establish a number of necessary conditions. Our insights also led us to efficient separation procedures and a branch-and-cut algorithm for the lifted multicut problem.

6 TREE PARTITIONS

6.1 INTRODUCTION

In this chapter we study the problem of partitioning a tree with respect to costs attributed to pairs of nodes, so as to minimize the sum of costs for those pairs of nodes that are in the same component. On the one hand, this *tree partition problem* is naturally formulated as the minimization of a multi-linear polynomial which exhibits a sparsity pattern that is determined by the underlying tree. On the other hand, the problem can be formulated as a lifted multicut problem (cf. Chapter 5) where the multicuts of the underlying tree are lifted to the complete graph. Therefore, we analyze the lifted multicut problem and its associated polyhedral geometry in this extreme case. In Section 6.2, we introduce the tree partition problem formally and discuss equivalent representations. While the multicut problem (P_{MC}) on trees is trivial, we observe that the tree partition problem is NP-hard. If the tree is a simple path, then the path partition problem is known to be polynomial time solvable (Kernighan, 1971). In Section 6.3 we study the lifted multicut polytope associated with the tree partition problem. We characterize several classes of facets for the general case, which in particular yields a complete totally dual integral description for the path case. Our study extends the results from Chapter 5 to the extreme case of minimally connected graphs such as trees and paths.

6.2 TREE PARTITION PROBLEM

Let $T = (V, E)$ be a tree. We use the short-hand notation $n = |E|$ and $m = |V|$ for the number of edges and nodes, respectively. For any pair of distinct nodes $u, v \in V$ denote by P_{uv} the unique path from u to v in T. Moreover, we denote by $d(u, v)$ the distance of u and v in T, i.e. the length of P_{uv}. We define a partition problem on T as the minimization of a particular multi-linear polynomial over binary inputs.

Definition 6.1. Let $T = (V, E)$ be a tree and $\bar{\theta} \in \mathbb{R}^{\binom{V}{2}}$ a vector associated with the node pairs. The optimization problem

$$\min_{y \in \{0,1\}^E} \sum_{uv \in \binom{V}{2}} \bar{\theta}_{uv} \prod_{e \in P_{uv}} y_e \tag{6.1}$$

is called the instance of the *tree partition problem* w.r.t. T and $\bar{\theta}$. If T is a path, then we also refer to (6.1) as the *path partition problem* w.r.t. T and $\bar{\theta}$.

Apparently, problem (6.1) corresponds to the minimzation of a certain class of pseudo-boolean functions (PBF). More precisely, we call any n-variate PBF *tree-sparse*, if its multi-linear polynomial form can be aligned with a tree such that every non-zero coefficient corresponds to the edge set of a path in the tree. Similary, we call it *path-sparse* if the tree is a path itself. Tree-sparse PBFs are exactly those PBFs that correspond to tree partition problems (6.1).

The tree partition problem can equivalently be written as a lifted multicut problem as follows. Denote by $\mathsf{LMC}(T) = \mathsf{LMC}(T, K)$ the lifted multicut polytope w.r.t. T and the complete graph K on V. Since T does not have any cycles and there is a unique path P_{uv} between any pair of distinct nodes $u, v \in V$, it follows from Lemma 5.2 that

$$\begin{aligned}\mathsf{LMC}(T) = \operatorname{conv}\Big\{x \in \{0,1\}^m \mid x_{uv} \le \sum_{e \in P_{uv}} x_e \qquad & \forall u, v \in V, d(u,v) \ge 2,\\ x_e \le x_{uv} \qquad & \forall u, v \in V, d(u,v) \ge 2,\ \forall e \in P_{uv}\Big\}.\end{aligned}$$

Lemma 6.1. *The vector $y \in \{0,1\}^n$ is a solution of problem (6.1) w.r.t. the tree $T = (V, E)$ and costs $\bar\theta \in \mathbb{R}^{\binom{V}{2}}$ if, and only if, the unique $x \in \mathsf{LMC}(T)$ such that $x_e = 1 - y_e$ for all $e \in E$ is a solution of the lifted multicut problem ($\mathrm{P}_{\mathsf{LMC}}$) w.r.t. T and the cost vector $\theta = -\bar\theta$.*

Proof. For any distinct pair of nodes $u, v \in V$, we introduce a binary variable $x_{uv} \in \{0,1\}$ via

$$x_{uv} = 1 - \prod_{e \in P_{uv}} y_e$$

which implies

$$x_{uv} = 0 \iff \forall e \in P_{uv}\colon\ y_e = 1 \iff \forall e \in P_{uv}\colon\ x_e = 0. \tag{6.2}$$

Therefore, we can reformulate problem (6.1) in terms of the variables x_{uv} by transforming the objective function according to

$$\bar\theta_{uv} \prod_{e \in P_{uv}} y_e = -\bar\theta_{uv}\big(1 - \prod_{e \in P_{uv}} y_e\big) + \bar\theta_{uv} = -\bar\theta_{uv}\, x_{uv} + \bar\theta_{uv}.$$

This leads to the linear combinatorial optimization problem

$$\min_{x \in \mathsf{LMC}(T)} \sum_{uv \in \binom{V}{2}} \theta_{uv}\, x_{uv} + \bar\theta_{uv},$$

where the definition of $\mathsf{LMC}(T)$ captures the relationship (6.2). □

The tree partition problem is NP-hard in general (Lemma 6.2 below). This shows that the lifted multicut problem w.r.t. T introduces additional complexity compared to the multicut problem w.r.t. T, since the multicut problem w.r.t. T is trivial. However, the path partition problem is polynomial time solvable (Kernighan, 1971).

Lemma 6.2. *The tree partition problem is NP-hard. It remains NP-hard if T is a star.*

Proof. If T is a star with n leaves, then problem (6.1) is equivalent to the unconstrained binary quadratic program with n variables, which is well-known to be NP-hard. □

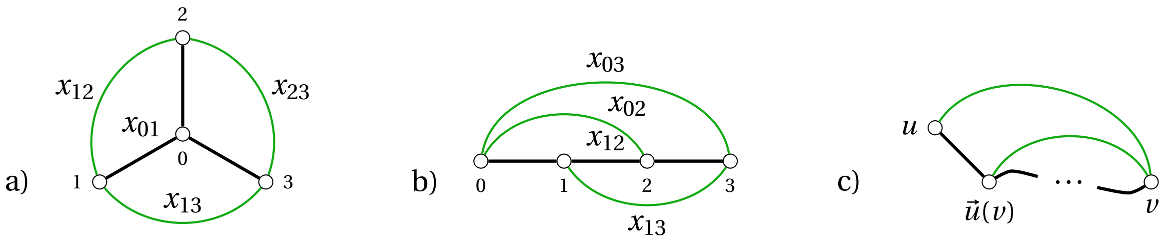

Figure 6.1: a) A star with additional (green, curved) edges between non-neighboring nodes corresponding to non-local variables. b) A path of length 3 lifted to the complete graph. c) The node $\vec{u}(v)$ is the first node on the path P_{uv}.

6.3 LIFTED MULTICUT POLYTOPE FOR TREES

In this section we study the lifted multicut polytope $\mathrm{LMC}(T)$ in greater detail. We characterize all trivial facets and offer an outer relaxation of $\mathrm{LMC}(T)$ that is tighter than the standard relaxation. In Section 6.3.2, we show that our results yield a complete totally dual integral (TDI) description of the lifted multicut polytope for paths.

We denote the standard outer relaxation of $\mathrm{LMC}(T)$ by

$$\begin{aligned}\Delta_T^0 = \Big\{x \in [0,1]^m \mid x_{uv} \le \sum_{e \in P_{uv}} x_e \qquad & \forall u, v \in V, d(u,v) \ge 2,\\ x_e \le x_{uv} \qquad & \forall u, v \in V, d(u,v) \ge 2,\ \forall e \in P_{uv}\Big\}.\end{aligned}$$

Let $\vec{u}(v)$ be the first node on the path P_{uv} that is different from both u and v (cf. Figure 6.1c) and consider the polytope

$$\begin{aligned}\Delta_T^1 = \Big\{x \in [0,1]^m \mid x_{uv} \le x_{u,\vec{u}(v)} + x_{\vec{u}(v),v} \qquad & \forall u, v \in V,\ d(u,v) \ge 2,\\ x_{\vec{u}(v),v} \le x_{uv} \qquad & \forall u, v \in V,\ d(u,v) \ge 2\Big\}.\end{aligned}$$

This description is canonical in the sense that it only considers a quadratic number of node triplets, namely those which feature two neighboring nodes and an arbitrary third node. The following lemma states that Δ_T^1 is indeed an outer relaxation of $\mathrm{LMC}(T)$ that is at least as tight as Δ_T^0.

Lemma 6.3. *It holds that* $\mathrm{LMC}(T) \subseteq \Delta_T^1 \subseteq \Delta_T^0$.

Proof. We show first that $\mathrm{LMC}(T) \subseteq \Delta_T^1$. For this purpose, let $x \in \mathrm{LMC}(T) \cap \mathbb{Z}^m$ be a vertex of $\mathrm{LMC}(T)$. If $x_{uv} > x_{u,\vec{u}(v)} + x_{\vec{u}(v),v}$ for some $u, v \in V$, then $x_{uv} = 1$ and $x_{u,\vec{u}(v)} = x_{\vec{u}(v),v} = 0$. This contradicts the fact that x satisfies all cut inequalities w.r.t. $\vec{u}(v), v$ and the path inequality w.r.t. u, v. If $x_{\vec{u}(v),v} > x_{uv}$ for some $u, v \in V$, then $x_{\vec{u}(v),v} = 1$ and $x_{uv} = 0$. This contradicts the fact that x satisfies all cut inequalities w.r.t. u, v and the path inequality w.r.t. $\vec{u}(v), v$. It follows that $x \in \Delta_T^1$.

Now, we show that $\Delta_T^1 \subseteq \Delta_T^0$. Let $x \in \Delta_T^1$. We need to show that x satisfies all path and cut inequalities. Let $u, v \in V$ with $d(u,v) \ge 2$. We proceed by induction on $d(u,v)$. If $d(u,v) = 2$, then the path and cut inequalities are directly given by the definition of

Δ_T^1 (for the two possible orderings of u and v). If $d(u,v) > 2$, then the path inequality is obtained from $x_{uv} \le x_{u,\vec{u}(v)} + x_{\vec{u}(v),v}$ and the induction hypothesis for the pair $\vec{u}(v), v$, since $d(\vec{u}(v), v) = d(u,v) - 1$. Similarly, for any edge e on the path from u to v, we obtain the cut inequality w.r.t. e by using the induction hypothesis and $x_{\vec{u}(v),v} \le x_{uv}$ such that (w.l.o.g.) e is on the path from $\vec{u}(v)$ to v. It follows that $x \in \Delta_T^0$. □

6.3.1 Facets

Lemma 6.4. *The inequality*

$$x_{uv} \le x_{u,\vec{u}(v)} + x_{\vec{u}(v),v} \tag{6.3}$$

for some $u, v \in V$ defines a facet of $\mathsf{LMC}(T)$ *if, and only if,* $d(u,v) = 2$.

Proof. First, suppose $d(u,v) = 2$. Then P_{uv} is a path of length 2 and thus chordless in the complete graph on V. Hence, the facet-defining property follows directly from Theorem 5.4. Now, suppose $d(u,v) > 2$ and let $x \in \mathsf{LMC}(T)$ be such that (6.3) is satisfied with equality. We show that this implies

$$x_{uv} + x_{\vec{u}(v),\vec{v}(u)} = x_{u,\vec{v}(u)} + x_{\vec{u}(v),v}. \tag{6.4}$$

Then the face of $\mathsf{LMC}(T)$ induced by (6.3) has dimension at most $m-2$ and hence cannot be a facet. In order to check that (6.4) holds, we distinguish the following three cases. If $x_{uv} = x_{u,\vec{u}(v)} = x_{\vec{u}(v),v}$, then all terms in (6.4) vanish. If $x_{uv} = x_{u,\vec{u}(v)} = 1$ and $x_{\vec{u}(v),v} = 0$, then $x_{\vec{u}(v),\vec{v}(u)} = 0$ and $x_{u,\vec{v}(u)} = 1$, so (6.4) holds. Finally, if $x_{uv} = x_{\vec{u}(v),v} = 1$ and $x_{u,\vec{u}(v)} = 0$, then (6.4) holds as well, because $x_{\vec{u}(v),\vec{v}(u)} = x_{u,\vec{v}(u)}$ by contraction of the edge $u, \vec{u}(v)$. □

Lemma 6.5. *The inequality*

$$x_{\vec{u}(v),v} \le x_{uv} \tag{6.5}$$

for some $u, v \in V$ defines a facet of $\mathsf{LMC}(T)$ *if, and only if, v is a leaf of T.*

Proof. First, suppose v is not a leaf of T and let $x \in \mathsf{LMC}(T)$ be such that (6.5) is satisfied with equality. Since v is not a leaf, there exists a neighbor $w \in V$ of v such that $P_{\vec{u}(v),v}$ is a subpath of $P_{\vec{u}(v),w}$ We show that x additionally satisfies the equality

$$x_{uw} = x_{\vec{u}(v),w} \tag{6.6}$$

and thus the face of $\mathsf{LMC}(T)$ induced by (6.5) cannot be a facet. There are two possible cases: Either $x_{uv} = x_{\vec{u}(v),v} = 1$, then $x_{uw} = x_{\vec{u}(v),w} = 1$ as well, or $x_{uv} = x_{\vec{u}(v),v} = 0$, then $x_{uw} = x_{vw} = x_{\vec{u}(v),w}$ by contraction of the path P_{uv}, so (6.6) holds.

Now, suppose v is a leaf of T and let Σ be the face of $\mathsf{LMC}(T)$ induced by (6.5). We need to prove that Σ has dimension $m-1$. This can be done explicitly by showing that we can construct $m-1$ distinct indicator vectors $y^{ww'}$ for $w, w' \in V$ as linear combinations of elements from the set $S = \{x \in \mathsf{LMC}(T) \cap \mathbb{Z}^m \mid x_{uv} = x_{\vec{u}(v),v}\}$.

This construction is analogous to the one used in the proof of Theorem 5.1. The difference here is that the vector $x^{\vec{u}(v),v} \notin S$, so we omit it from the construction (and thus omit $y^{\vec{u}(v),v}$ as well). Thus, we end up with $m-1$ vectors y^e for $e \in \binom{V}{2} \setminus \{\vec{u}(v)v\}$ that are linearly independent and constructed as linear combinations of $\mathbb{1} - x^e \in S$, so the claim follows. □

Lemma 6.6. *For any distinct $u, v \in V$, the inequality $x_{uv} \leq 1$ defines a facet of* $\mathsf{LMC}(T)$ *if, and only if, both u and v are leaves of T. Moreover, none of the inequalities $0 \leq x_{uv}$ define facets of* $\mathsf{LMC}(T)$.

Proof. We apply the more general characterization given by Theorems 5.3 and 5.2. The nodes $u, v \in V$ are a pair of ww'-cut-vertices for some vertices $w, w' \in V$ (with at least one being different from u and v) if, and only if, u or v is not a leaf of V. Thus, the claim follows from Theorem 5.2. The second assertion follows from Theorem 5.3 and the fact that we lift to the complete graph on V. □

We present another large class of non-trivial facets of $\mathsf{LMC}(T)$. For any $u, v \in V$ with $d(u, v) \geq 3$ consider the inequality

$$x_{uv} + x_{\vec{u}(v),\vec{v}(u)} \leq x_{u,\vec{v}(u)} + x_{\vec{u}(v),v}, \tag{6.7}$$

which we refer to as *square inequality*. As an example consider the graph depicted in Figure 6.1b with $u = 0$ and $v = 3$.

Lemma 6.7. *Any square inequality is valid for* $\mathsf{LMC}(T)$.

Proof. Let $x \in \mathsf{LMC}(T) \cap \mathbb{Z}^m$ and suppose that either $x_{u,\vec{v}(u)} = 0$ or $x_{\vec{u}(v),v} = 0$ for some $u, v \in V$ with $d(u, v) \geq 3$. Then, since x satisfies all cut inequalities w.r.t. $u, \vec{v}(u)$, respectively $\vec{u}(v), v$, and the path inequality w.r.t. $\vec{u}(v), \vec{v}(u)$, it must hold that $x_{\vec{u}(v),\vec{v}(u)} = 0$. Moreover, if even $x_{u,\vec{v}(u)} = 0 = x_{\vec{u}(v),v}$, then, by the same reasoning, we have $x_{uv} = 0$ as well. Hence, x satisfies (6.7). □

Lemma 6.8. *Any square inequality defines a facet of* $\mathsf{LMC}(T)$.

Proof. Let Σ be the face of $\mathsf{LMC}(T)$ induced by (6.7) for some $u, v \in V$ with $d(u, v) \geq 3$. We need to prove that Σ has dimension $m - 1$, which can be done explicitly by showing that we can construct $m - 1$ distinct indicator vectors $y^{w,w'}$ for $w, w' \in V$ as linear combinations of elements from the set $S = \{x \in \mathsf{LMC}(T) \cap \mathbb{Z}^m \mid x_{uv} + x_{\vec{u}(v),\vec{v}(u)} = x_{u,\vec{v}(u)} + x_{\vec{u}(v),v}\}$.

Again, the construction is very similar to the one used in the proof of Theorem 5.1. Observe that for any feasible $x \in \mathsf{LMC}(T) \cap \mathbb{Z}^m$ with $x \notin S$, we must have that $x_{\vec{u}(v),\vec{v}(u)} = 0$ and $x_{u,\vec{u}(v)} = x_{\vec{v}(u),v} = 1$. This means that the vector $x^{\vec{u}(v),\vec{v}(u)} \notin S$ in the construction. We simply omit this vector and similarly $y^{\vec{u}(v),\vec{v}(u)}$ from the construction. By the same reasoning, we conclude that the m constructed vectors $\{\mathbb{1}\} \cup \left\{x^e \mid e \in \binom{V}{2} \setminus \{\vec{u}(v)\vec{v}(u)\}\right\}$ are affine independent, so the claim follows. □

We note that further facets can be established by the connection of $\mathsf{LMC}(T)$ to the multi-linear polytope and, as a special case, the *Boolean Quadric Polytope* (Padberg, 1989). This is, however, beyond the scope of this thesis and we focus here on facets that are associated with paths in the tree T.

6.3.2 Lifted Multicut Polytope for Paths

In this section we show that the facets established in the previous section yield a complete description of $\mathsf{LMC}(P)$, where P is a path. To this end, suppose that $V = \{0,\dots,n\}$ and $E = \Big\{\{i,i+1\} \mid i \in \{0,\dots,n-1\}\Big\}$ are linearly ordered. Therefore, $P = (V,E)$ is path. We consider only paths of length $n \geq 2$, since for $n = 1$, the polytope $\mathsf{LMC}(P) = [0,1]$ is simply the unit interval. Let Δ_P^{Path} be the polytope defined by

$$\begin{aligned}
\Delta_P^{\text{Path}} = \big\{x \in \mathbb{R}^m \mid x_{0n} &\leq 1, && (6.8)\\
x_{in} &\leq x_{i-1,n} && \forall i \in \{1,\dots,n-1\}, \quad (6.9)\\
x_{0i} &\leq x_{0,i+1} && \forall i \in \{1,\dots,n-1\}, \quad (6.10)\\
x_{i-1,i+1} &\leq x_{i-1,i} + x_{i,i+1} && \forall i \in \{1,\dots,n-1\}, \quad (6.11)\\
x_{j,k} + x_{j+1,k-1} &\leq x_{j+1,k} + x_{j,k-1} && \forall j,k \in \{0,\dots,n\},\ j<k-2\big\}. \quad (6.12)
\end{aligned}$$

Note that the system of inequalities (6.8) – (6.12) consists precisely of those inequalities which we have shown to define facets of $\mathsf{LMC}(P)$ in the previous section. We first prove that Δ_P^{Path} indeed yields an outer relaxation of $\mathsf{LMC}(P)$.

Lemma 6.9. *It holds that* $\mathsf{LMC}(P) \subseteq \Delta_P^{\text{Path}} \subseteq \Delta_P^1$.

Proof. First, we show that $\mathsf{LMC}(P) \subseteq \Delta_P^{\text{Path}}$. Let $x \in \mathsf{LMC}(P) \cap Z^m$, then x satisfies (6.8) and (6.11) by definition. Suppose x violates (6.9), then $x_{in} = 1$ and $x_{i-1,n} = 0$. This contradicts the fact that x satisfies all cut inequalities w.r.t. $i-1,n$ and the path inequality w.r.t. i,n. So, x must satisfy (6.9) and, by symmetry, also (6.10). It follows from Lemma 6.7 that x satisfies (6.12) as well and thus $x \in \Delta_P^{\text{Path}}$.

Next, we prove that $\Delta_P^{\text{Path}} \subseteq \Delta_P^1$. To this end, let $x \in \Delta_P^{\text{Path}}$. We show that x satisfies all inequalities (6.5). Let $u,v \in V$ with $u < v-1$. We need to prove that both $x_{u+1,v} \leq x_{uv}$ and $x_{u,v-1} \leq x_{uv}$ hold. For reasons of symmetry, it suffices to show only $x_{u+1,v} \leq x_{uv}$. We proceed by induction on the distance of u from n. If $v = n$, then $x_{u+1,n} \leq x_{un}$ is given by (6.9). Otherwise, we use (6.12) for $j = u$ and $k = v+1$ and the induction hypothesis on $v+1$:

$$\begin{aligned}
x_{uv} + x_{u+v,v+1} &\geq x_{u+1,v} + x_{u,v+1}\\
&\geq x_{u+1,v} + x_{u+1,v+1}\\
\implies x_{uv} &\geq x_{u+1,v}.
\end{aligned}$$

It remains to show that x satisfies all inequalities (6.3). Let $u,v \in V$ with $u < v-1$. We proceed by induction on $d(u,v) = u-v$. If $d(u,v) = 2$, then (6.3) is given by (6.11). If $d(u,v) > 2$, then we use (6.12) for $j = u$ and $k = v$ as well as the induction hypothesis on $u, v-1$, which have distance $d(u,v)-1$:

$$\begin{aligned}
x_{uv} + x_{u+1,v-1} &\leq x_{u+1,v} + x_{u,v-1}\\
&\leq x_{u+1,v} + x_{u,u+1} + x_{u+1,v-1}\\
\implies x_{uv} &\leq x_{u,u+1} + x_{u+1,v}.
\end{aligned}$$

Hence, $x \in \Delta_P^1$, which concludes the proof. □

As our main result in this chapter, we prove that Δ_P^{Path} is in fact a complete description of $\mathsf{LMC}(P)$. To derive this, we utilize the notion of total dual integrality. For an extensive reference on this subject we refer the reader to (Schrijver, 1986).

Definition 6.2. A system of linear inequalities $Ax \leq b$ with $A \in \mathbb{Q}^{k\times m}$, $b \in \mathbb{Q}^k$ is called *totally dual integral* (TDI) if for any $c \in \mathbb{Z}^m$ such that the linear program $\max\{c^\top x \mid Ax \leq b\}$ is feasible and bounded, there exists an integral optimal dual solution.

Total dual integrality is an important concept in polyhedral geometry as it serves as a sufficient condition on the integrality of polyhedra according to the following fact.

Fact 6.1 ((Edmonds and Giles, 1977))**.** If $Ax \leq b$ is totally dual integral and b is integral, then the polytope defined by $Ax \leq b$ is integral.

Theorem 6.1. *The system* (6.8) – (6.12) *is totally dual integral.*

Proof. We rewrite system (6.8) – (6.12) more compactly in the following way. Introduce two artificial nodes $-\infty$ and ∞ where we associate $-\infty$ to any index less than 0 and ∞ to any index greater than n. Moreover, we introduce variables x_{ii} for all $0 \leq i \leq n$ as well as $x_{-\infty,i}$ and $x_{i,\infty}$ for all $1 \leq i \leq n-1$ and finally $x_{-\infty,\infty}$. Now, the system (6.8) – (6.12) is equivalent to the system

$$x_{j,k} + x_{j+1,k-1} \leq x_{j+1,k} + x_{j,k-1} \qquad \forall j,k \in \{-\infty, 0, \ldots, n, \infty\},\ j \leq k-2 \tag{6.13}$$

given the additional equality constraints

$$x_{ii} = 0 \quad \forall 0 \leq i \leq n, \tag{6.14}$$
$$x_{-\infty,i} = 1 \quad \forall 1 \leq i \leq n-1, \tag{6.15}$$
$$x_{i,\infty} = 1 \quad \forall 1 \leq i \leq n-1, \tag{6.16}$$
$$x_{-\infty,\infty} = 1. \tag{6.17}$$

Let the system defined by (6.13) – (6.17) be represented in matrix form as $Ax \leq a$, $Bx = b$. Note that Δ_P^{Path} is non-empty and bounded. Thus, to establish total dual integrality, we need to show that for any $\theta \in \mathbb{Z}^{m+3n}$ the dual program

$$\min\{a^\top y + b^\top z \mid A^\top y + B^\top z = \theta, y \geq 0\} \tag{6.18}$$

has an integral optimal solution. Here, the y variables, indexed by j,k, correspond to the inequalities (6.13) and the z variables, indexed by pairs of $i, -\infty$ and ∞, correspond to the equations (6.14) – (6.17). Then, the equation system $A^\top y + B^\top z = \theta$ translates to

$$y_{i-1,i+1} + z_{i,i} = \theta_{i,i} \tag{6.19}$$
$$y_{i-1,i+2} - y_{i-1,i+1} - y_{i,i+2} = \theta_{i,i+1} \tag{6.20}$$
$$y_{i-1,\ell+1} - y_{i-1,\ell} - y_{i,\ell+1} + y_{i,\ell} = \theta_{i,\ell} \tag{6.21}$$
$$-y_{-\infty,i+1} + y_{-\infty,i} + z_{-\infty,i} = \theta_{-\infty,i} \tag{6.22}$$
$$-y_{i-1,\infty} + y_{i,\infty} + z_{i,\infty} = \theta_{i,\infty} \tag{6.23}$$
$$y_{-\infty,\infty} + z_{-\infty,\infty} = \theta_{-\infty,\infty}, \tag{6.24}$$

where (6.19) – (6.21) hold for all $0 \le i < i+1 < \ell \le n$ and (6.22), (6.23) hold for all $1 \le i \le n-1$. Observe that (6.19) includes only y variables with indices of distance 2, (6.20) couples y variables of distance 3 with those of distance 2 and (6.21) couples the remaining y variables of any distance $d > 3$ with those of distance $d-1$ and $d-2$. Hence, any choice of values for the free variables z_{ii} completely determines all y variables. This means we can eliminate y and reformulate the dual program entirely in terms of the z variables, as follows. It holds that

$$\begin{aligned} &0 \le y_{i-1,i+1} = \theta_{i,i} - z_{i,i} && \forall 0 \le i \le n, \\ &0 \le y_{i-1,i+2} = \theta_{i,i+1} + \theta_{i,i} + \theta_{i+1,i+1} - z_{i,i} - z_{i+1,i+1} && \forall 0 \le i < i+1 \le n \end{aligned}$$

and thus, by (6.21),

$$0 \le y_{i-1,\ell+1} = \sum_{i \le j \le k \le \ell} \theta_{j,k} - \sum_{k=i}^{\ell} z_{k,k}$$

for all $0 \le i \le \ell \le n$. Substituting the y variables in (6.22) – (6.24) yields the following equivalent formulation of the dual program (6.18):

$$\begin{aligned} \min \quad & z_{-\infty,\infty} + \sum_{i=0}^{n} z_{-\infty,i} + z_{i,\infty} && (6.25) \\ \text{s.t.} \quad & \sum_{k=i}^{\ell} z_{k,k} \le \sum_{i \le j \le k \le \ell} \theta_{j,k} && \forall 0 \le i \le \ell \le n \\ & z_{-\infty,i} + z_{i,i} = \theta_{-\infty,i} + \sum_{0 \le j \le i} \theta_{j,i} && \forall 1 \le i \le n-1 \\ & z_{i,\infty} + z_{i,i} = \theta_{i,\infty} + \sum_{i \le k \le n} \theta_{i,k} && \forall 1 \le i \le n-1 \\ & z_{-\infty,\infty} - \sum_{k=1}^{n} z_{k,k} = \theta_{-\infty,\infty} - \sum_{0 \le j \le k \le n} \theta_{j,k}. \end{aligned}$$

The variables $z_{-\infty,i}$, $z_{i,\infty}$ and $z_{-\infty,\infty}$ occur only in a single equation each. Furthermore, the matrix corresponding to the inequality constraints satisfies the *consecutive-ones* property. Therefore, the constraint matrix of the whole system is totally unimodular, which concludes the proof. □

Remark. The constraint matrix corresponding to the system (6.8) – (6.12) is in general *not* totally unimodular. A minimal example is the path of length 4.

Corollary 6.1. *It holds that* $\mathsf{LMC}(P) = \Delta_P^{\text{Path}}$.

Proof. This is immediate from Lemma 6.9, Fact 6.1 and Theorem 6.1. □

Note that the path partition problem admits an even more efficient representation as a set partitioning problem as follows. For each $0 \le i \le \ell \le n$, let

$$d_{i,\ell} = \sum_{0 \le i \le j \le k \le n} \theta_{j,k},$$

then taking the dual of problem (6.25) and simplifying yields the problem

$$\begin{aligned} \min \quad & d^\top \lambda \\ \text{s.t.} \quad & \sum_{0 \le i \le k \le \ell \le n} \lambda_{i,\ell} = 1, \quad \forall 0 \le k \le n \\ & \lambda \ge 0. \end{aligned} \tag{6.26}$$

Each variable $\lambda_{i,\ell}$ corresponds to the component containing nodes i to ℓ. Problem (6.26) is precisely the sequential set partitioning formulation of the path partition problem as used by Joseph and Bryson (1997). It admits a quadratic number of variables and a linear number of constraints (opposed to a quadratic number of constraints in the description of $\mathsf{LMC}(P)$). The integrality constraint need not be enforced, since the constraint matrix is totally unimodular.

6.4 CONCLUSION

We studied the lifted multicut polytope for the special cases of trees lifted to the complete graph. We characterize a number of its facets and provide a tighter relaxation compared to the standard linear relaxation. The described facets provide a complete totally dual integral description of the associated lifted multicut polytope for paths. This result relates the geometry of the path partition problem to the combinatorial properties of alternative formulations such as the sequential set partitioning problem.

7 MORAL LINEAGE TRACING

7.1 INTRODUCTION

RECENT advances in microscopy have enabled biologists to observe organisms on a cellular level with higher spatio-temporal resolution than before Chen et al. (2014); Greenbaum et al. (2012); Tomer et al. (2012). Analysis of such microscopy sequences is key to several open questions in biology, including embryonic development of complex organisms Keller et al. (2010, 2008), tissue formation Guillot and Lecuit (2013) or the understanding of metastatic behavior of tumor cells Zervantonakis et al. (2012). However, to get from a sequence of raw microscopy images to biologically or clinically relevant quantities, such as cell motility, migration patterns and differentiation schedules, robust methods for *cell lineage tracing* are required and have therefore received considerable attention Amat et al. (2014, 2013); Chenouard et al. (2014); Li et al. (2008); Maška et al. (2014); Meijering et al. (2012).

Cell lineage tracing is typically considered a two step problem: In the first step, individual cells are detected and segmented in every image. Then, in the second step, individual cells are tracked over time and, in case of a cell division, linked to their ancestor cell, to finally arrive at the lineage forest of all cells (Figure 7.1). The tracking subproblem is complicated by cells that enter or leave the field of view, or low temporal resolution that allows large displacements or even multiple consecutive divisions within one time step. In addition to this, mistakes made in the first step, leading to over- or undersegmentation of the cells, propagate into the resulting lineage forest and cause spurious divisions or missing branches, respectively. The tracking subproblem is closely related to multi-target tracking Berclaz et al. (2011); Tang et al. (2015); Wang et al. (2014); Insafutdinov et al. (2017); Tang et al. (2017) or reconstruction of tree-like structures Funke et al. (2012); Rempfler et al. (2015, 2016); Türetken et al. (2016); Türetken et al. (2011). It has been cast in the form of different optimization problems Jug et al. (2014); Kausler et al. (2012); Padfield et al. (2011); Schiegg et al. (2015, 2013) that can deal with some of the mentioned difficulties, e.g., by selecting from multiple segmentation hypotheses Schiegg et al. (2015, 2013).

Jug et al. (2016), on the other hand, have proposed a rigorous mathematical abstraction of the joint problem which they call the *moral lineage tracing problem* (MLTP). It is a hybrid of the lifted multicut problem, and the *minimum cost disjoint arborescence* problem, variations of which have been applied to reconstruct lineage forests in Jug et al. (2014); Kausler et al. (2012); Padfield et al. (2011); Schiegg et al. (2013, 2015) or tree-like structures Funke et al. (2012); Türetken et al. (2011); Türetken et al. (2016). Feasible solutions to the MLTP define not only a valid cell lineage forest over time, but also a segmentation of the cells in every frame (cf. Figure 7.2). Solving this optimization problem therefore tackles both subtasks – segmentation and tracking – simultaneously. While Jug et al. (2016) demonstrate the advantages of their approach in terms of robustness, they also observe that their branch-and-cut algorithm (as well as the cutting-plane algorithm for the linear relaxation they study) is prone to a large number of cuts and exhibits slow convergence on large instances. That, unfortunately, prevents many applications of the MLTP in practice, since it would be too computationally expensive.

In this chapter, we improve the branch-and-cut algorithm from Jug et al. (2016) by separating tighter cutting planes and by employing specialized heuristics to extract feasible solutions. We demonstrate the superior convergence of our algorithm on the problem instances from Jug et al. (2016), solving two (previously unsolved) instances to optimality and obtaining accurate solutions orders of magnitude faster. We further indicate the scalability of our algorithm on larger (previously inaccessible) instances.

7.2 MORAL LINEAGE TRACING PROBLEM

Consider a set of $\mathcal{T} = \{0, \dots, t_{\text{end}}\}$ consecutive frames of microscopy image data. In moral lineage tracing, we seek to jointly segment the frames into cells and track the latter and their descendants over time. This problem is formulated by Jug et al. (2016) as an ILP with binary variables for all edges in an undirected graph as follows.

For each time index $t \in \mathcal{T}$, the node set V_t comprises all cell fragments, e.g. superpixels, in frame t. Each neighboring pair of cell fragments are connected by an edge. The collection of such edges is denoted by E_t. Between consecutive frames t and $t+1$, cell fragments that are sufficiently close to each other are connected by a (temporal) edge. The set of such inter frame edges is denoted by $E_{t,t+1}$. By convention, we set $V_{t_{\text{end}}+1} = E_{t_{\text{end}}+1} = E_{t_{\text{end}},t_{\text{end}}+1} = \emptyset$. The graph $G = (V, E)$ with $V = \bigcup_{t \in \mathcal{T}} V_t$ and $E = \bigcup_{t \in \mathcal{T}} (E_t \cup E_{t,t+1})$ is called *hypothesis graph* and illustrated in Figure 7.2. For convenience, we further write $G_t = (V_t, E_t)$ for the subgraph corresponding to frame t and $G_{t,t+1} = (V_t \cup V_{t+1}, E_t \cup E_{t,t+1} \cup E_{t+1})$ for the subgraph corresponding to frames t and $t+1$.

For any hypothesis graph $G = (V, E)$, a set $L \subseteq E$ is called a *lineage cut* of G and, correspondingly, the subgraph $(V, E \setminus L)$ is called a *lineage subgraph* of G if

i. For every $t \in \mathcal{T}$, the set $E_t \cap L$ is a multicut of G_t.

ii. For every $t \in \mathcal{T}$ and every $uv \in E_{t,t+1} \cap L$, the nodes u and v are not path-connected in the graph $G_{t,t+1} \setminus L$.

iii. For every $t \in \mathcal{T}$ and nodes $u_t, v_t \in V_t$, $u_{t+1}, v_{t+1} \in V_{t+1}$ with $u_t u_{t+1}, v_t v_{t+1} \in E_{t,t+1} \setminus L$ and such that u_{t+1} and v_{t+1} are path-connected in $(V, E_{t+1} \setminus L)$, the nodes u_t and v_t are path-connected in $(V, E_t \setminus L)$.

For any lineage graph $(V, E \setminus L)$ and every $t \in \mathcal{T}$, the non-empty, maximal connected subgraphs of $(V_t, E_t \setminus L)$ are called *cells* at time index t. Furthermore, Jug et al. (2016) call a lineage cut, respectively lineage graph, *binary* if it additionally satisfies

iv. For every $t \in \mathcal{T}$, every cell at time t is connected to at most two distinct cells at time $t+1$.

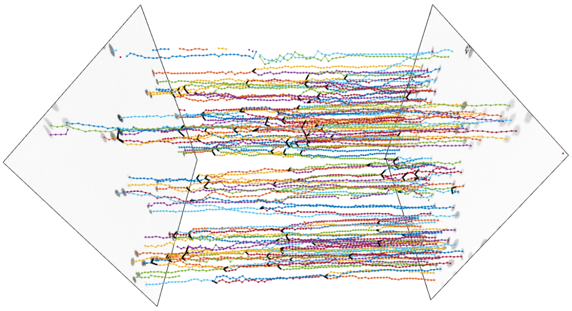

Figure 7.1: Depicted above is a lineage forest of cells from a sequence of microscopy images. The first image of the sequence is shown on the left. The last image is shown on the right. Cell divisions are depicted in black.

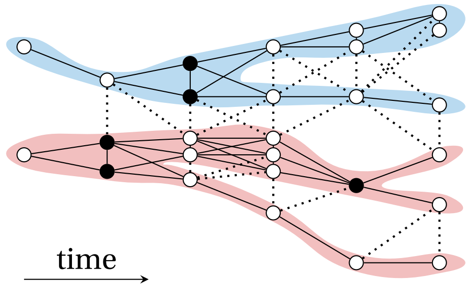

Figure 7.2: The *moral lineage tracing problem* (MLTP)[1]: Given a sequence of images decomposed into cell fragments (depicted as nodes in the figure), cluster fragments into cells in each frame and *simultaneously* associate cells into lineage forests over time. Solid edges indicate joint cells within images and descendant relations across images. Black nodes depict fragments of cells about to divide.

According to Jug et al. (2016), any lineage graph well-defines a lineage forest of cells. Moreover, a lineage cut (and thus a lineage graph) can be encoded as a 01-labeling on the edges of the hypothesis graph.

Lemma 7.1 (Jug et al. (2016))**.** *For every hypothesis graph $G = (V, E)$ and every $x \in \{0,1\}^E$, the set of edges $e \in E$ such that $x_e = 1$ is a lineage cut of G if, and only if, x satisfies inequalities* (7.1) – (7.3)*:*

$$\forall t \in \mathcal{T}\ \forall \text{ cycles } C \text{ of } G_t\ \forall f \in C: \qquad x_f \leq \sum_{e \in C \setminus \{f\}} x_e \tag{7.1}$$

$$\forall t \in \mathcal{T}\ \forall uv \in E_{t,t+1}\ \forall uv\text{-paths } P \text{ in } G_{t,t+1}: \qquad x_{uv} \leq \sum_{e \in P} x_e \tag{7.2}$$

$$\forall t \in \mathcal{T}\ \forall u_t u_{t+1}, v_t v_{t+1} \in E_{t,t+1} (\text{with } u_t, v_t \in V_t)$$
$$\forall u_t v_t\text{-cuts } \delta(U) \text{ in } G_t\ \forall u_{t+1} v_{t+1}\text{-paths } P \text{ in } G_{t+1}:$$
$$1 - \sum_{e \in \delta(U)} (1 - x_e) \leq x_{u_t u_{t+1}} + x_{v_t v_{t+1}} + \sum_{e \in P} x_e \tag{7.3}$$

Jug et al. (2016) refer to (7.1) as *space cycle*, to (7.2) as *space-time cycle* and to (7.3) as *morality* constraints. We denote by X'_G the set of all $x \in \{0,1\}^E$ that satisfy (7.1) – (7.3).

[1]The figure is a correction of the one displayed in (Jug et al., 2016).

For the formulation of the additional *bifurcation* constraints, which guarantee that the associated lineage cut is binary, we refer to (Jug et al., 2016, Eq. 4). The set X_G collects all $x \in X'_G$ that also satisfy the bifurcation constraints.

Given cut costs $\theta \in \mathbb{R}^E$ on the edges as well as *birth* and *termination* costs $\theta^+, \theta^- \in \mathbb{R}^V_+$ on the vertices of the hypothesis graph, Jug et al. (2016) define the following *moral lineage tracing problem* (MLTP)

$$\min_{x,x^+,x^-} \quad \sum_{e \in E} \theta_e x_e + \sum_{v \in V} \theta_v^+ x_v^+ + \sum_{v \in V} \theta_v^- x_v^- \tag{7.4}$$

$$\text{s.t.} \quad x \in X_G, \quad x^+, x^- \in \{0,1\}^V, \tag{7.5}$$

$$\forall t \in \mathcal{T} \, \forall v \in V_{t+1} \forall V_t v\text{-cuts } \delta(U) \text{ in } G_{t,t+1}: \quad 1 - x_v^+ \le \sum_{e \in \delta(U)} (1 - x_e), \tag{7.6}$$

$$\forall t \in \mathcal{T} \, \forall v \in V_t \forall v V_{t+1}\text{-cuts } \delta(U) \text{ in } G_{t,t+1}: \quad 1 - x_v^- \le \sum_{e \in \delta(U)} (1 - x_e). \tag{7.7}$$

The inequalities (7.6) and (7.7) are called *birth* and *termination* constraints, respectively.

7.3 IMPROVED BRANCH-AND-CUT ALGORITHM

Jug et al. (2016) propose to solve the MLTP with a branch-and-cut algorithm, for which they design separation procedures for inequalities (7.1) – (7.3), (7.6) – (7.7) and the bifurcation constraints. In the following, we propose several modifications of the optimization algorithm, which drastically improve its performance.

Clearly, it is sufficient to consider only chordless cycles for the inequalities (7.1), since chordal cycle inequalities are dominated by chordless cycle inequalities (cf. Chapter 5). This argument can be analogously transferred to inequalities (7.2) and (7.3).

Moreover, those inequalities of (7.3) where $u_t v_t \in E_t$ is an edge of the hypothesis graph may be considerably strengthened by a less trivial, yet simple modification. In total, Lemma 7.2 shows that with both results combined, we can equivalently replace (7.1) – (7.3) by the simpler and tighter set of inequalities (7.8) and (7.9). In relation to our improved version of the branch-and-cut algorithm, we refer to (7.8) as *cycle* and to (7.9) as *morality* constraints.

Lemma 7.2. *For every hypothesis graph $G = (V, E)$ it holds that $x \in X'_G$ if, and only if, we have that $x \in \{0,1\}^E$ and x satisfies*

$$\forall t \in \mathcal{T} \, \forall uv \in E_t \cup E_{t,t+1} \forall \text{ chordless } uv\text{-paths } P \text{ in } G_{t,t+1}: \quad x_{vw} \le \sum_{e \in P} x_e \tag{7.8}$$

$$\forall t \in \mathcal{T} \, \forall u', v' \in V_t \text{ such that } u'v' \notin E_t$$
$$\forall u'v'\text{-cuts } \delta(U) \text{ in } G_t \forall \text{ chordless } u'v'\text{-paths } P \text{ in } G_{t,t+1}: \quad 1 - \sum_{e \in \delta(U)} (1 - x_e) \le \sum_{e \in P} x_e \tag{7.9}$$

Proof. We first show that any $x \in \{0,1\}^E$ satisfying all of (7.1) – (7.3) also satisfies (7.8) and (7.9) by contraposition. First, assume $x \in \{0,1\}^E$ violates an inequality of (7.8) for some $t \in \mathcal{T}$, $uv \in E_t \cup E_{t,t+1}$ and chordless uv-path P. We distinguish the following cases: If

$uv \in E_t$ and P is a path in G_t, then the inequality is included in (7.1). If $uv \in E_{t,t+1}$, then the inequality is included in (7.2). It remains to consider the case that $uv \in E_t$ and P is not entirely contained in G_t. Let $u_t u_{t+1}, v_t v_{t+1} \in E_{t,t+1}$ (with $u_t, v_t \in V_t$) be edges of P such that $v_t v_{t+1}$ is the first inter-frame edge after $u_t u_{t+1}$. Furthermore, let $P_{u_{t+1} v_{t+1}} \subseteq E_{t+1}$ be the subpath of P between u_{t+1} and v_{t+1}. Now, either there is a $u_t v_t$-cut $\delta(U)$ in G_t such that $x_{\delta(U)} = 1$ or there is a $u_t v_t$-path P' in G_t such that $x_{P'} = 0$. In the first case this yields an inequality of (7.3) corresponding to $\delta(U), u_t u_{t+1}, v_t v_{t+1}$ and $P_{u_{t+1} v_{t+1}}$ violated by x. In the second case we can replace $P_{u_{t+1} v_{t+1}}$ by P' and repeat the argument for another pair of inter-frame edges in P. If this process never results in a violated morality constraint, then we finally obtain an inequality of (7.1) corresponding to $uv \cup P$ that is violated by x.

Next, suppose $x \in \{0,1\}^E$ violates an inequality of (7.9) for some $t \in \mathcal{T}$, $u', v' \in V_t$ with $u'v' \notin E_t$, some $u'v'$-cut in G_t and a chordless $u'v'$-path P in $G_{t,t+1}$. Similar to the last paragraph we can find that x violates an inequality of (7.3) corresponding to $\delta(U), u_t u_{t+1}, v_t v_{t+1}$ and $P_{u_{t+1} v_{t+1}}$, defined analogously.

For the converse, we show that if $x \in \{0,1\}^E$ satisfies the inequalities (7.8) and (7.9), then it also satisfies (7.1) – (7.3). Any cycle in $G_{t,t+1}$ which is not chordless can be split into two cycles contained in $G_t, G_{t,t+1}$ or G_{t+1} which share exactly one edge. Therefore, any inequality of (7.1) – (7.2) is implied by a combination of inequalities from (7.8). A similar argument shows that any inequality from (7.3) for $u_t v_t \notin E_t$ is implied by a combination of inequalities from (7.8) and (7.9). Finally, if $u_t v_t \in E_t$, then for any $u_t v_t$-cut $\delta(U)$ in G_t it holds that $u_t v_t \in \delta(U)$. Thus, we can reapply the previous reasoning together with the simple fact that

$$1 - \sum_{e \in \delta(U)} (1 - x_e) \leq 1 - (1 - x_{u_t v_t}) = x_{u_t v_t}$$

to obtain the same result. □

Note that the MLTP is closely related to the lifted multicut problem in the following sense. Suppose we introduce for every pair of non-neighboring nodes $u', v' \in V_t$ a variable $x_{u'v'}$ indicating whether u' and v' belong to the same cell ($x_{u'v'} = 0$) or not ($x_{u'v'} = 1$). Then the additional variables make connectedness of cells via future frames explicit. Moreover, any inequality of (7.9) is exactly the combination of a *cut inequality* $1 - x_{u'v'} \leq \sum_{e \in \delta(U)} (1 - x_e)$ and a *path inequality* $x_{u'v'} \leq \sum_{e \in P} x_e$ in the sense of lifted multicuts. For neighboring nodes $u, v \in V_t$, i.e. $uv \in E_t$, we have the variable x_{uv} at hand and can thus omit the cut part of the morality constraint, as the lemma shows.

Termination and Birth Constraints We further suggest a strengthening of the birth and termination constraints in the MLTP. To this end, for any $v \in V_{t+1}$ let $V_t(v) = \{u \in V_t \mid uv \in E_{t,t+1}\}$ be the set of neighboring nodes in frame t. Further, we denote by $E\big(V_t(v), V_{t+1} \setminus \{v\}\big)$ the set of inter frame edges that connect some node $u_t \in V_t(v)$ with some node $u_{t+1} \in V_{t+1}$ different from v.

Lemma 7.3. *For every hypothesis graph $G = (V,E)$, the vectors $x \in X'_G$, $x^+, x^- \in \{0,1\}^V$ satisfy inequalities (7.6) if, and only if,*

$$\forall t \in \mathcal{T}\, \forall v \in V_{t+1} \forall V_t v\text{-cuts } \delta(U) \text{ in } G_{t,t+1} : 1 - x_v^+ \leq \sum_{e \in \delta(U) \setminus E(V_t(v), V_{t+1} \setminus \{v\})} (1 - x_e). \qquad (7.10)$$

Similarly, $x \in X'_G$, $x^+, x^- \in \{0,1\}^V$ *satisfy* (7.7) *if, and only if*

$$\forall t \in \mathcal{T}\, \forall v \in V_t \forall v V_{t+1}\text{-cuts}\, \delta(U) \text{ in } G_{t,t+1} : 1 - x^-_v \leq \sum_{e \in \delta(U) \setminus E(V_t \setminus \{v\}, V_{t+1}(v))} (1 - x_e). \quad (7.11)$$

Proof. We show the claim only for birth constraints since the proof for termination constraints is analogous. Let $x \in X'_G$ and $x^+, x^- \in \{0,1\}^V$. Apparently, if (7.10) is satisfied, then

$$\sum_{e \in \delta(U) \setminus E(V_t \setminus \{v\}, V_{t+1}(v))} (1 - x_e) \leq \sum_{e \in \delta(U)} (1 - x_e)$$

implies that (7.6) also holds. Conversely, suppose (7.10) is violated. Then there exists some $t \in \mathcal{T}$, $v \in V_{t+1}$ and a $V_t v$-cut$\delta(U)$ in $G_{t,t+1}$ such that $x^+_v = 0$ and $x_e = 1$ for all $e \in \delta(U) \setminus E\big(V_t(v), V_{t+1} \setminus \{v\}\big)$. Assume (7.6) is not violated, then there is a path P in $G_{t,t+1}$ from some node in V_t to v with $x_P = 0$. Then P must have non-empty intersection with $E\big(V_t(v), V_{t+1} \setminus \{v\}\big)$. Let $u \in V_t(v)$ and $v' \in V_{t+1} \setminus \{v\}$ be such that $uv' \in P$. Since $x_{uv} = 1$ it follows that x violates the inequality

$$x_{uv} \leq \sum_{e \in P_{uv}} x_e$$

of (7.2) where P_{uv} is the subpath of P from u to v. This is a contradiction to $x \in X'_G$. □

Additional Odd Wheel Constraints We propose to further include additional odd wheel inequalities (cf. Section 1.2.3) in the MLTP in order to strengthen the corresponding LP relaxation. More precisely, we consider only wheels $W = (\{v_1, \dots, v_{|C|}\} \cup \{u\}, C \cup E_u) \subset G$ such that the center node $u \in V_{t+1}$ for some $t \in \mathcal{T}$ and $v_i \in V_t$ for all $i \in \{1, \dots, |C|\}$. This structure guarantees that for any $x \in X'_G$, the restriction $x_{C \cup E_u}$ is the characteristic vector of a multicut of W. Therefore, the vector x satisfies the associated wheel inequality.

Implementation For a subset of the constraints, we use the commercial branch-and-cut solver Gurobi to solve the LP relaxation and find integer feasible solutions. Whenever Gurobi finds an integer feasible solution x, we check whether $x \in X_G$ and all birth and termination constraints are satisfied. If not, then we provide Gurobi with an additional batch of violated inequalities from (7.8) – (7.11) as well as violated bifurcation constraints and repeat. To this end, we adapt the separation procedures of Jug et al. (2016) to account for our improvements in a straight-forward manner. We further add odd wheel inequalities for wheels with 3 outer nodes as described above (so-called *3-wheels*) to the starting LP relaxation.

For every integer feasible solution that Gurobi finds, we fix the connected components of the intra-frame segmentation and solve the remaining problem heuristically (Rempfler et al., 2017). This allows for the early extraction of feasible lineage forests from the ILP.

7.4 EXPERIMENTS

Instances and Setup We evaluate our branch-and-cut algorithm on the two large instances from Jug et al. (2016): *Flywing-epithelium* and *N2DL-HeLa-full*. The hypothesis

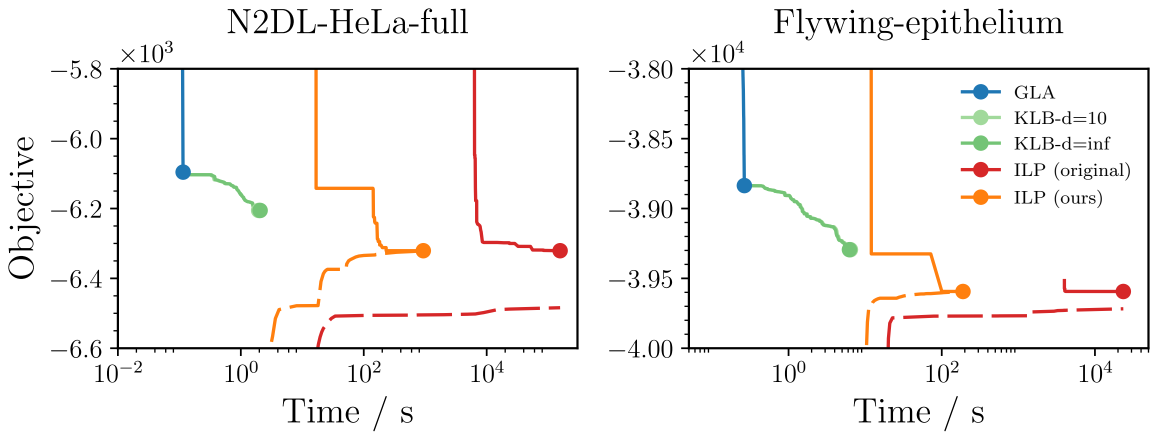

Figure 7.3: Comparison of algorithms for the MLTP in terms of runtime, objective (solid) and bounds (dashed) on the large instances of Jug et al. (2016). Our branch-and-cut algorithm (ILP ours) converges to the optimal solution in up to one hundredth of the time of the original branch-and-cut algorithm (ILP original) and provides tight bounds in both cases. For comparison, we also plot the objective values obtained by the heuristic algorithms (GLA, KLB) from Rempfler et al. (2017).

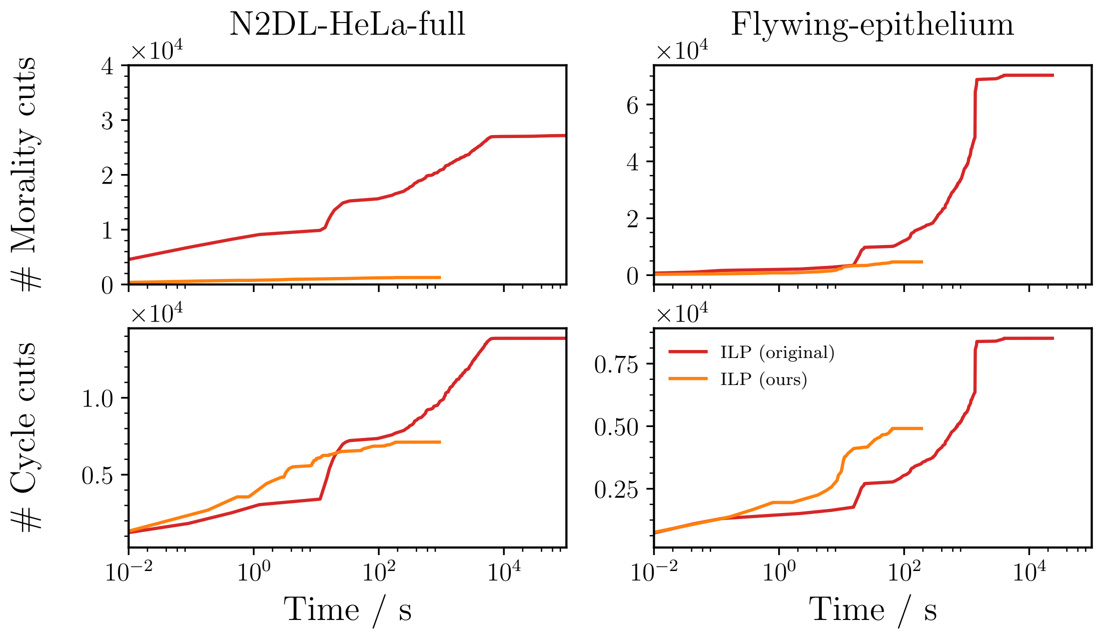

Figure 7.4: Number of morality cuts (**top**), i.e. (7.3) or (7.9), and cycle cuts (**bottom**), i.e. (7.1) and (7.2) or (7.8), separated in the different branch-and-cut algorithms. We observe that our branch-and-cut algorithm requires considerably fewer morality cuts, while the number of cycle cuts (including both space-cycles and space-time-cycles) is in the same order of magnitude.

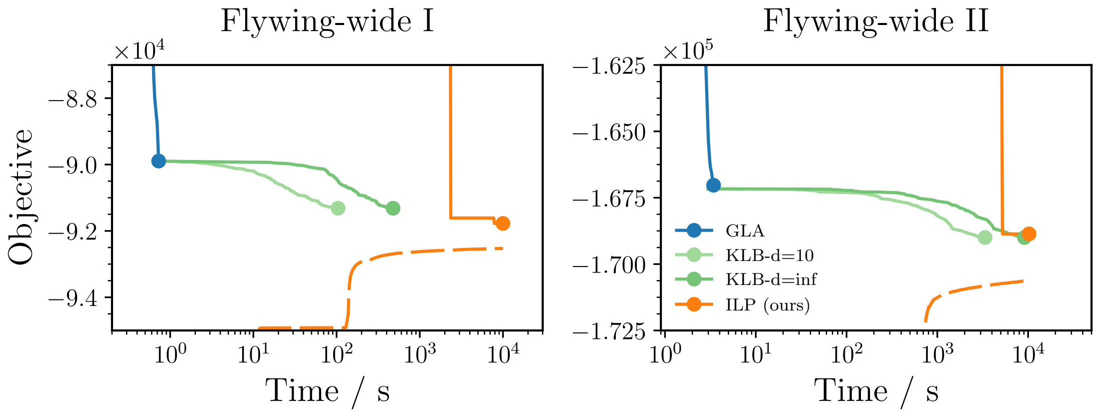

Figure 7.5: Runtime results on the more extensive instances *Flywing-wide I* and *II*.

graph of the former instance consists of 5026 nodes and 19011 edges, while the latter consists of 10882 nodes and 19807 edges. In addition to this, we report experiments on two more sequences of a flywing epithelium time-lapse microscopy with a wider field of view. Their hypothesis graphs consist of 10641 nodes and 42236 edges, respectively 76747 edges. We denote the data sets with *Flywing-wide I* and *II*. These instances are preprocessed with the same pipeline as *Flywing-epithelium*. For details on the preprocessing and the choice of birth and termination costs, we refer to Jug et al. (2016).

Convergence Analysis The improved branch-and-cut algorithm retrieves feasible solutions considerably faster and provides tighter bounds than the algorithm from Jug et al. (2016), as can be seen in Figure 7.3. The instances *Flywing-epithelium* and *N2DL-HeLa* are solved to optimality in less than 200s, respectively 1000s, while the original algorithm did not find any feasible solutions in that time. As shown in Figure 7.4, we observe that our modifications of the branch-and-cut algorithm greatly reduce the number of morality cuts. For the larger instances *Flywing-wide I* and *II*, we compare the runtime of our algorithm to the heuristics GLA and KLB (Rempfler et al., 2017) in Figure 7.5.

7.5 CONCLUSION

For the moral lineage tracing problem (MLTP), we improved the branch-and-cut algorithm of Jug et al. (2016) by separating tighter cutting planes and employing specialized heuristics in the branch-and-bound search. The improvements rely on an analysis of the polytope associated with the MLTP, which is closely related to the lifted multicut polytope. Our branch-and-cut algorithm solves previous instances quickly to optimality. Moreover, our algorithm is applicable to larger instances as well. This demonstrates the practical benefit that comes with a better understanding of the polyhedral geometry of the MLTP.

8 CONCLUSION

In this thesis we have studied multicuts in two ways. In the first part, we developed optimization guarantees for the multicut problem. In the second part, we analysed the polyhedral geometry associated with lifted multicuts.

The optimization guarantees we considered in the first part are i) nontrivial lower bounds that are available fast, ii) partial optimality certificates that can be computed efficiently, and iii) tightness characterizations of linear programming relaxations. We devised a heuristic algorithm for the dual of the multicut problem to quickly produce lower bounds even for large instances that are not amenable to standard linear programming. We have shown that for practial instances the bounds often remain fairly tight and, moreover, we can obtain better primal solutions by exploiting the reduced costs of the dual solutions. In order to identify parts of optimal solutions, we developed combinatorial persistency criteria and devised efficient algorithms to check them. Our approach includes elementary criteria that can be evaluated enumeratively as well as more sophisticated criteria that require fast primal and dual heuristics. In combination, our method is able to reduce the size of practical instances considerably. This shows that the problem exhibits substantial structural simplicity in practice. We investigated this observation from another, more theoretical point of view and analysed the sign patterns of the cost vector that prohibit tightness of the cycle relaxation. We are able to characterize its tightness for the special cases of trees and circuits, and further provide necessary conditions in the general case.

The results in the first part are relevant for the theory and practice of multicut optimization. Our dual heuristic algorithm may be used as an initialization for more sophisticated methods that solve the dual or even as a standalone tool to compute lower bounds fast. The persistency method is an effective preprocessing technique, which is paramount in the context of global optimization for large instances. It remains open whether more effective criteria alongside efficient algorithms for their evaluation can be found. Our analysis of the cycle relaxation tightness hints at the challenges associated with deriving a characterization for general graphs, which remains an open theoretical research problem.

In the second part we studied the lifted multicut polytope and relatives thereof. For the general case, we analysed the fundamental inequalities associated with lower and upper variable bounds as well as cycles and cuts. For each inequality type, we derived conditions under which the inequality defines a facet of the lifted multicut polytope. The conditions are necessary and sometimes sufficient for box inequalities. For cycle inequalities we gave an exact characterization and for cut inequalities we established a number of necessary conditions. When restricted to trees, we observed that the lifted multicut problem reduces the minimization of a sparse multi-linear polynomial over binary variables. We gave a tighter description of the lifted multicut polytope in this special case, which is even complete for simple paths. However, our results in particular show that the lifted multicut problem can be NP-hard even if the multicut problem on the base graph is trivial. For the moral lineage tracing problem, which is a close relative of the lifted multicut problem, we provide a tighter description of its feasible set and thus improve the performance of a branch-and-cut algorithm from prior work.

The results in the second part are relevant for branch-and-cut algorithms that solve an integer linear programming formulation of the lifted multicut problem or related variants. Our findings help in understanding the complexity of the problem and are vital when solving it to optimality. It is apparent that the connectivity constraints introduce an intricate structure to the problem and unfortunately the cut inequalities are rarely facet-defining. In order to improve the performance of branch-and-cut algorithms further, it would be interesting to derive other families of inequalities that i) are stronger than the cut inequalities and ii) can be separated efficiently.

LIST OF FIGURES

LIST OF TABLES

A. Abdi, A. E. Feldmann, B. Guenin, J. Könemann, and L. Sanità. Lehman's theorem and the directed steiner tree problem. *SIAM Journal on Discrete Mathematics*, 30(1):141–153, 2016. doi: 10.1137/15M1007185. 46

W. P. Adams, J. B. Lassiter, and H. D. Sherali. Persistency in 0-1 polynomial programming. *Mathematics of Operations Research*, 23(2):359–389, 1998. ISSN 0364765X, 15265471. 24

N. Ailon, M. Charikar, and A. Newman. Aggregating inconsistent information: Ranking and clustering. *J. ACM*, 55(5):23:1–23:27, 2008. doi: 10.1145/1411509.1411513. 10

N. Ailon, N. Avigdor-Elgrabli, E. Liberty, and A. van Zuylen. Improved approximation algorithms for bipartite correlation clustering. *SIAM Journal on Computing*, 41(5):1110–1121, 2012. doi: 10.1137/110848712. 10

A. Alush and J. Goldberger. Ensemble segmentation using efficient integer linear programming. *TPAMI*, 34(10):1966–1977, 2012. doi: 10.1109/TPAMI.2011.280. 24, 28, 41

F. Amat, E. W. Myers, and P. J. Keller. Fast and robust optical flow for time-lapse microscopy using super-voxels. *Bioinformatics*, 29(3):373–380, 2013. 93

F. Amat, W. Lemon, D. P. Mossing, K. McDole, Y. Wan, K. Branson, E. W. Myers, and P. J. Keller. Fast, accurate reconstruction of cell lineages from large-scale fluorescence microscopy data. *Nature methods*, 2014. 93

B. Andres, J. H. Kappes, T. Beier, U. Köthe, and F. A. Hamprecht. Probabilistic image segmentation with closedness constraints. In *ICCV*, 2011. 12, 13

A. Bailoni, C. Pape, S. Wolf, T. Beier, A. Kreshuk, and F. A. Hamprecht. A generalized framework for agglomerative clustering of signed graphs applied to instance segmentation. *CoRR*, abs/1906.11713, 2019. 12

E. Balas and C. H. Martin. Pivot and complement-a heuristic for 0-1 programming. *Management Science*, 26(1):86–96, 1980. 39

N. Bansal, A. Blum, and S. Chawla. Correlation clustering. *Machine Learning*, 56(1–3): 89–113, 2004. doi: 10.1023/B:MACH.0000033116.57574.95. 9, 10, 17

T. Beier, T. Kroeger, J. H. Kappes, U. Köthe, and F. A. Hamprecht. Cut, glue & cut: A fast, approximate solver for multicut partitioning. In *CVPR*, 2014. 12, 17

T. Beier, F. A. Hamprecht, and J. H. Kappes. Fusion moves for correlation clustering. In *CVPR*, pages 3507–3516, 2015. doi: 10.1109/CVPR.2015.7298973. 12

T. Beier, B. Andres, U. Köthe, and F. A. Hamprecht. An efficient fusion move algorithm for the minimum cost lifted multicut problem. In *ECCV*, pages 715–730, 2016. doi: 10.1007/978-3-319-46475-6_44. 59

T. Beier, C. Pape, N. Rahaman, T. Prange, S. Berg, D. D. Bock, A. Cardona, G. W. Knott, S. M. Plaza, L. K. Scheffer, U. Koethe, A. Kreshuk, and F. A. Hamprecht. Multicut brings automated neurite segmentation closer to human performance. *Nature Methods*, 14(2): 101–102, 2017. doi: 10.1038/nmeth.4151. 12, 17, 23, 39, 59

C. Bentz, M.-C. Costa, L. Létocart, and F. Roupin. Multicuts and integral multiflows in rings. *European Journal of Operational Research*, 196(3):1251 – 1254, 2009. doi: https://doi.org/10.1016/j.ejor.2008.05.006. 46

J. Berclaz, F. Fleuret, E. Türetken, and P. Fua. Multiple object tracking using k-shortest paths optimization. *IEEE transactions on pattern analysis and machine intelligence*, 33 (9):1806–1819, 2011. 93

C. Berge. Balanced matrices. *Mathematical Programming*, 2(1):19–31, Feb 1972. doi: 10.1007/BF01584535. 50

E. Boros and P. L. Hammer. Pseudo-boolean optimization. *Discrete Applied Mathematics*, 123(1–3):155 – 225, 2002. doi: https://doi.org/10.1016/S0166-218X(01)00341-9. 24

E. Boros, P. L. Hammer, R. Sun, and G. Tavares. A max-flow approach to improved lower bounds for quadratic unconstrained binary optimization (qubo). *Discrete Optimization*, 5(2):501 – 529, 2008. doi: https://doi.org/10.1016/j.disopt.2007.02.001. In Memory of George B. Dantzig. 24

Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *TPAMI*, 26(9):1124–1137, 2004. doi: 10.1109/TPAMI.2004.60. 39

N. Cesa-Bianchi, C. Gentile, F. Vitale, and G. Zappella. A correlation clustering approach to link classification in signed networks. In *COLT*, 2012. 12, 17

M. Charikar, V. Guruswami, and A. Wirth. Clustering with qualitative information. *Journal of Computer and System Sciences*, 71(3):360–383, 2005. doi: 10.1016/j.jcss.2004.10.012. 10

S. Chawla, R. Krauthgamer, R. Kumar, Y. Rabani, and D. Sivakumar. On the hardness of approximating multicut and sparsest-cut. *Computational Complexity*, 15(2):94–114, 2006. doi: 10.1007/s00037-006-0210-9. 10

S. Chawla, K. Makarychev, T. Schramm, and G. Yaroslavtsev. Near optimal LP rounding algorithm for correlation clustering on complete and complete k-partite graphs. In *STOC*, 2015. doi: 10.1145/2746539.2746604. 10

B.-C. Chen, W. R. Legant, K. Wang, L. Shao, D. E. Milkie, M. W. Davidson, C. Janetopoulos, X. S. Wu, J. A. Hammer, Z. Liu, et al. Lattice light-sheet microscopy: Imaging molecules to embryos at high spatiotemporal resolution. *Science*, 346(6208):1257998, 2014. 93

N. Chenouard, I. Smal, F. De Chaumont, M. Maska, I. F. Sbalzarini, Y. Gon, J. Cardinale, C. Carthel, S. Coraluppi, M. Winter, A. R. Cohen, W. J. Godinez, K. Rohr, Y. Kalaidzidis, L. Liang, J. Duncan, H. Shen, Y. Xu, K. Magnusson, J. Jalden, H. M. Blau, P. Paul-Gilloteaux, P. Roudot, C. Kervrann, F. Waharte, J.-Y. Tinevez, S. L. Shorte, J. Willemse, K. Celler, G. P. Van Wezel, H.-W. Dan, Y.-S. Tsai, C. Ortiz De Solorzano, J.-C. Olivo-Marin, and E. Meijering. Objective comparison of particle tracking methods. *Nature Methods*, 11(3):281–289, 2014. 93

S. Chopra. The graph partitioning polytope on series-parallel and 4-wheel free graphs. *SIAM Journal on Discrete Mathematics*, 7(1):16–31, 1994. doi: 10.1137/S0895480191199415. 8, 45

S. Chopra and M. Rao. The partition problem. *Mathematical Programming*, 59(1–3):87–115, 1993. doi: 10.1007/BF01581239. 5, 7, 8, 11, 59, 61, 66, 81

J. Cohen, L. A. Rodrigues, F. Silva, R. Carmo, A. L. P. Guedes, and E. P. Duarte. Parallel implementations of gusfield's cut tree algorithm. In *Algorithms and Architectures for Parallel Processing*, pages 258–269, 2011. 39

D. Cornaz. Max-multiflow/min-multicut for G+H series-parallel. *Discrete Mathematics*, 311 (17):1957–1967, 2011. doi: 10.1016/j.disc.2011.05.025. 45, 46, 48

G. Cornuéjols and B. Novick. Ideal 0, 1 matrices. *J. Comb. Theory, Ser. B*, 60(1):145–157, 1994. doi: 10.1006/jctb.1994.1009. 46, 54

G. Cornuéjols, B. Guenin, and L. Tunçel. Lehman matrices. *J. Comb. Theory, Ser. B*, 99(3): 531–556, 2009. doi: 10.1016/j.jctb.2008.06.009. 53

E. D. Demaine, D. Emanuel, A. Fiat, and N. Immorlica. Correlation clustering in general weighted graphs. *Theoretical Computer Science*, 361(2–3):172–187, 2006. doi: 10.1016/j.tcs.2006.05.008. 8, 9, 10, 17

J. Desmet, M. De Maeyer, B. Hazes, and I. Lasters. The dead-end elimination theorem and its use in protein side-chain positioning. *Nature*, 356(6369):539, 1992. 24

M. Deza, M. Grötschel, and M. Laurent. Complete descriptions of small multicut polytopes. In *Applied Geometry And Discrete Mathematics, Proceedings of a DIMACS Workshop*, pages 221–252, 1990. 8

M. M. Deza, M. Grötschel, and M. Laurent. Clique-web facets for multicut polytopes. *Mathematics of Operations Research*, 17(4):981–1000, 1992. doi: 10.1287/moor.17.4.981. 8

J. Edmonds and F. R. Giles. A min-max relation for submodular functions on graphs. *Annals of Discrete Mathematics*, 1:185 – 204, 1977. doi: http://dx.doi.org/10.1016/S0167-5060(08) 70734-9. 89

L. R. Ford and D. R. Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 8:399–404, 1956. doi: 10.4153/CJM-1956-045-5. 45

J. Funke, B. Andres, F. A. Hamprecht, A. Cardona, and M. Cook. Efficient automatic 3d-reconstruction of branching neurons from em data. In *CVPR*, pages 1004–1011. IEEE, 2012. 93

N. Garg, V. V. Vazirani, and M. Yannakakis. Primal-dual approximation algorithms for integral flow and multicut in trees. *Algorithmica*, 18(1):3–20, 1997. doi: 10.1007/BF02523685. 46, 50

R. E. Gomory and T. C. Hu. Multi-Terminal Network Flows. *Journal of the Society for Industrial and Applied Mathematics*, 9(4):551–570, 1961. ISSN 03684245. doi: 10.2307/2098881. 30

A. Greenbaum, W. Luo, T.-W. Su, Z. Göröcs, L. Xue, S. O. Isikman, A. F. Coskun, O. Mudanyali, and A. Ozcan. Imaging without lenses: achievements and remaining challenges of wide-field on-chip microscopy. *Nature methods*, 9(9):889–895, 2012. 93

I. Gridchyn and V. Kolmogorov. Potts model, parametric maxflow and k-submodular functions. In *ICCV*, 2013. 24

M. Grötschel and W. Pulleyblank. Weakly bipartite graphs and the max-cut problem. *Operations Research Letters*, 1(1):23 – 27, 1981. doi: https://doi.org/10.1016/0167-6377(81)90020-1. 45

M. Grötschel and Y. Wakabayashi. A cutting plane algorithm for a clustering problem. *Mathematical Programming*, 45(1):59–96, 1989. doi: 10.1007/BF01589097. 8, 9

M. Grötschel and Y. Wakabayashi. Facets of the clique partitioning polytope. *Mathematical Programming*, 47:367–387, 1990. doi: 10.1007/BF01580870. 8

B. Guenin. A characterization of weakly bipartite graphs. *J. Comb. Theory Ser. B*, 83(1):112–168, Sept. 2001. ISSN 0095-8956. doi: 10.1006/jctb.2001.2051. 45, 46

C. Guillot and T. Lecuit. Mechanics of epithelial tissue homeostasis and morphogenesis. *Science*, 340(6137):1185–1189, 2013. 93

D. Gusfield. Very simple methods for all pairs network flow analysis. *SIAM J. Comput.*, 19(1):143–155, 1990. doi: 10.1137/0219009. 39

P. L. Hammer, P. Hansen, and B. Simeone. Roof duality, complementation and persistency in quadratic 0–1 optimization. *Mathematical Programming*, 28(2):121–155, 1984. doi: 10.1007/BF02612354. 24

M. Henzinger, A. Noe, C. Schulz, and D. Strash. Practical minimum cut algorithms. In *Proceedings of the Twentieth Workshop on Algorithm Engineering and Experiments (ALENEX)*, pages 48–61. SIAM, 2018. 24, 25

A. Horňáková, J.-H. Lange, and B. Andres. Analysis and optimization of graph decompositions by lifted multicuts. In *ICML*, 2017. 3

T. C. Hu. Multi-commodity network flows. *Oper. Res.*, 11(3):344–360, 1963. doi: 10.1287/opre.11.3.344. 45

E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele. Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In *ECCV*, pages 34–50, 2016. doi: 10.1007/978-3-319-46466-4_3. 12, 17

E. Insafutdinov, M. Andriluka, L. Pishchulin, S. Tang, E. Levinkov, B. Andres, and B. Schiele. Arttrack: Articulated multi-person tracking in the wild. In *CVPR*, pages 1293–1301, 2017. doi: 10.1109/CVPR.2017.142. 12, 93

A. Joseph and N. Bryson. Partitioning of sequentially ordered systems using linear programming. *Computers & Operations Research*, 24(7):679 – 686, 1997. doi: http://dx.doi.org/10.1016/S0305-0548(96)00070-6. 91

F. Jug, T. Pietzsch, D. Kainmüller, J. Funke, M. Kaiser, E. van Nimwegen, C. Rother, and G. Myers. Optimal joint segmentation and tracking of escherichia coli in the mother machine. In *Bayesian and grAphical Models for Biomedical Imaging*, pages 25–36. Springer, 2014. 93

F. Jug, E. Levinkov, C. Blasse, E. W. Myers, and B. Andres. Moral lineage tracing. In *CVPR*, pages 5926–5935, 2016. doi: 10.1109/CVPR.2016.638. 4, 93, 94, 95, 96, 98, 99, 100

F. Kahl and P. Strandmark. Generalized roof duality. *Discrete Applied Mathematics*, 160 (16-17):2419–2434, 2012. doi: 10.1016/j.dam.2012.06.009. 24

S. Kahruman-Anderoglu, E. Kolotoglu, S. Butenko, and I. V. Hicks. On greedy construction heuristics for the MAX-CUT problem. *International Journal of Computational Science and Engineering (IJCSE)*, 3(3):211–218, 2007. doi: 10.1504/IJCSE.2007.017827. 39

J. H. Kappes, M. Speth, B. Andres, G. Reinelt, and C. Schnörr. Globally optimal image partitioning by multicuts. In *EMMCVPR*, 2011. 12, 13, 17

J. H. Kappes, B. Andres, F. A. Hamprecht, C. Schnörr, S. Nowozin, D. Batra, S. Kim, B. X. Kausler, T. Kröger, J. Lellmann, N. Komodakis, B. Savchynskyy, and C. Rother. A comparative study of modern inference techniques for structured discrete energy minimization problems. *IJCV*, 115(2):155–184, 2015. doi: 10.1007/s11263-015-0809-x. 11, 20, 21, 39

J. H. Kappes, M. Speth, G. Reinelt, and C. Schnörr. Higher-order segmentation via multicuts. *Computer Vision and Image Understanding*, 143:104–119, 2016a. doi: 10.1016/j.cviu.2015.11.005. 12, 59

J. H. Kappes, P. Swoboda, B. Savchynskyy, T. Hazan, and C. Schnörr. Multicuts and perturb & MAP for probabilistic graph clustering. *Journal of Mathematical Imaging and Vision*, 56(2):221–237, 2016b. doi: 10.1007/s10851-016-0659-3. 12

A. Kardoost and M. Keuper. Solving minimum cost lifted multicut problems by node agglomeration. In *ACCV*, pages 74–89, Cham, 2019. Springer International Publishing. 12

A. Karzanov. Polyhedra related to undirected multicommodity flows. *Linear Algebra and its Applications*, 114-115:293 – 328, 1989. doi: https://doi.org/10.1016/0024-3795(89)90468-0. Special Issue Dedicated to Alan J. Hoffman. 45

B. X. Kausler, M. Schiegg, B. Andres, M. Lindner, U. Koethe, H. Leitte, J. Wittbrodt, L. Hufnagel, and F. A. Hamprecht. A discrete chain graph model for 3d+ t cell tracking with high misdetection robustness. In *ECCV*, pages 144–157. Springer, 2012. 93

P. J. Keller, A. D. Schmidt, J. Wittbrodt, and E. H. Stelzer. Reconstruction of zebrafish early embryonic development by scanned light sheet microscopy. *Science*, 322(5904):1065–1069, 2008. ISSN 0036-8075. doi: 10.1126/science.1162493. URL http://science.sciencemag.org/content/322/5904/1065. 93

P. J. Keller, A. D. Schmidt, A. Santella, K. Khairy, Z. Bao, J. Wittbrodt, and E. H. Stelzer. Fast, high-contrast imaging of animal development with scanned light sheet-based structured-illumination microscopy. *Nature methods*, 7(8):637–642, 2010. 93

B. W. Kernighan. Optimal sequential partitions of graphs. *J. ACM*, 18(1):34–40, Jan. 1971. ISSN 0004-5411. doi: 10.1145/321623.321627. 83, 84

M. Keuper. Higher-order minimum cost lifted multicuts for motion segmentation. In *ICCV*, pages 4252–4260, 2017. doi: 10.1109/ICCV.2017.455. 12, 59

M. Keuper, B. Andres, and T. Brox. Motion trajectory segmentation via minimum cost multicuts. In *ICCV*, 2015a. doi: 10.1109/ICCV.2015.374. 12, 17

M. Keuper, E. Levinkov, N. Bonneel, G. Lavoué, T. Brox, and B. Andres. Efficient decomposition of image and mesh graphs by lifted multicuts. In *ICCV*, 2015b. doi: 10.1109/ICCV.2015.204. 12, 17, 39, 59, 80

S. Kim, C. D. Yoo, S. Nowozin, and P. Kohli. Image segmentation using higher-order correlation clustering. *TPAMI*, 36(9):1761–1774, 2014. doi: 10.1109/TPAMI.2014.2303095. 12, 59

A. Kirillov, E. Levinkov, B. Andres, B. Savchynskyy, and C. Rother. Instancecut: From edges to instances with multicut. In *CVPR*, pages 7322–7331, 2017. doi: 10.1109/CVPR.2017.774. 13

P. N. Klein, C. Mathieu, and H. Zhou. Correlation clustering and two-edge-connected augmentation for planar graphs. In *STACS*, 2015. doi: 10.4230/LIPIcs.STACS.2015.554. 10

P. Kohli and P. H. S. Torr. Effciently solving dynamic markov random fields using graph cuts. In *ICCV*, pages 922–929, 2005. doi: 10.1109/ICCV.2005.81. 39

P. Kohli, A. Shekhovtsov, C. Rother, V. Kolmogorov, and P. Torr. On partial optimality in multi-label MRFs. In *ICML*, 2008. 24

V. Kolmogorov. Generalized roof duality and bisubmodular functions. *Discrete Applied Mathematics*, 160(4):416 – 426, 2012. doi: https://doi.org/10.1016/j.dam.2011.10.026. 24

I. Kovtun. Partial optimal labeling search for a np-hard subclass of (max,+) problems. In B. Michaelis and G. Krell, editors, *Pattern Recognition*, pages 402–409, 2003. ISBN 978-3-540-45243-0. 24

I. Kovtun. Sufficient condition for partial optimality for (max,+)-labeling problems and its usage. *Control systems and machines*, (2):35–42, 2011. 24

J. Lange. Flow-partitionable signed graphs. *CoRR*, abs/2005.01536, 2020. 3

J. Lange and B. Andres. Decomposition of trees and paths via correlation. *CoRR*, abs/1706.06822, 2017. 4

J.-H. Lange, A. Karrenbauer, and B. Andres. Partial optimality and fast lower bounds for weighted correlation clustering. In *ICML*, 2018. 3, 10, 24, 25, 35, 41

J.-H. Lange, B. Andres, and P. Swoboda. Combinatorial persistency criteria for multicut and max-cut. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3

A. Lehman. On the width - length inequality. *Math. Program.*, 17(1):403–417, 1979. doi: 10.1007/BF01588263. 46, 47

A. Lehman. The width-length inequality and degenerate projective planes. In *Polyhedral Combinatorics, Proceedings of a DIMACS Workshop, Morristown, New Jersey, USA, June 12-16, 1989*, pages 101–106, 1990. 46, 47

J. Leskovec, D. P. Huttenlocher, and J. M. Kleinberg. Signed networks in social media. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems, CHI*, pages 1361–1370, 2010. doi: 10.1145/1753326.1753532. 20, 21

E. Levinkov, A. Kirillov, and B. Andres. A comparative study of local search algorithms for correlation clustering. In *GCPR*, 2017. doi: 10.1007/978-3-319-66709-6_9. 12, 17, 21

K. Li, E. D. Miller, M. Chen, T. Kanade, L. E. Weiss, and P. G. Campbell. Cell population tracking and lineage construction with spatiotemporal context. *Medical image analysis*, 12(5):546–566, 2008. 93

F. Liers, M. Jünger, G. Reinelt, and G. Rinaldi. *Computing Exact Ground States of Hard Ising Spin Glass Problems by Branch-and-Cut*, chapter 4, pages 47–69. Wiley-Blackwell, 2005. doi: 10.1002/3527603794.ch4. 40

M. Maška, V. Ulman, D. Svoboda, P. Matula, P. Matula, C. Ederra, A. Urbiola, T. España, S. Venkatesan, D. M. W. Balak, P. Karas, T. Bolcková, M. Streitová, C. Carthel, S. Coraluppi, N. Harder, K. Rohr, K. E. G. Magnusson, J. Jaldén, H. M. Blau, O. Dzyubachyk, P. Křížek, G. M. Hagen, D. Pastor-Escuredo, D. Jimenez-Carretero, M. J. Ledesma-Carbayo, A. Muñoz-Barrutia, E. Meijering, M. Kozubek, and C. Ortiz-de Solorzano. A benchmark for comparison of cell tracking algorithms. *Bioinformatics*, 30(11):1609–1617, 2014. 93

K. Mehlhorn, A. Neumann, and J. M. Schmidt. Certifying 3-edge-connectivity. *Algorithmica*, 77(2):309–335, Feb 2017. doi: 10.1007/s00453-015-0075-x. 35

E. Meijering, O. Dzyubachyk, and I. Smal. Methods for cell and particle tracking. *Methods in Enzymology*, 504(9):183–200, 2012. 93

C. Michini. Forbidden minors for tight cycle relaxations. Technical report, 2016. 45

G. L. Nemhauser and L. E. Trotter. Vertex packings: structural properties and algorithms. *Mathematical Programming*, 8(1):232–248, 1975. 24

S. Nowozin and C. H. Lampert. Global interactions in random field models: A potential function ensuring connectedness. *SIAM Journal on Imaging Sciences*, 3(4):1048–1074, 2010. doi: 10.1137/090752614. 59

M. Padberg. The boolean quadric polytope: Some characteristics, facets and relatives. *Mathematical Programming*, 45(1):139–172, 1989. doi: 10.1007/BF01589101. 87

M. Padberg and G. Rinaldi. An efficient algorithm for the minimum capacity cut problem. *Mathematical Programming*, 47(1):19–36, May 1990. doi: 10.1007/BF01580850. 24

D. Padfield, J. Rittscher, and B. Roysam. Coupled minimum-cost flow cell tracking for high-throughput quantitative analysis. *Medical image analysis*, 15(4):650–668, 2011. 93

X. Pan, D. S. Papailiopoulos, S. Oymak, B. Recht, K. Ramchandran, and M. I. Jordan. Parallel correlation clustering on big graphs. In *NIPS*, 2015. 10

C. Pape, T. Beier, P. Li, V. Jain, D. D. Bock, and A. Kreshuk. Solving large multicut problems for connectomics via domain decomposition. In *ICCV Workshops*, 2017. doi: 10.1109/ICCVW.2017.7. 12, 39, 40

L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. V. Gehler, and B. Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. In *CVPR*, pages 4929–4937, 2016. doi: 10.1109/CVPR.2016.533. 12

M. Rempfler. *Graph Inference in Biomedical Image Analysis*. PhD thesis, Technical University of Munich, Germany, 2019. 4

M. Rempfler, M. Schneider, G. D. Ielacqua, X. Xiao, S. R. Stock, J. Klohs, G. Székely, B. Andres, and B. H. Menze. Reconstructing cerebrovascular networks under local physiological constraints by integer programming. *Medical Image Analysis*, 25(1):86 – 94, 2015. ISSN 1361-8415. doi: https://doi.org/10.1016/j.media.2015.03.008. 93

M. Rempfler, B. Andres, and B. Menze. The minimum cost connected subgraph problem in medical image analysis. In *MICCAI*, pages 397–405, 2016. doi: 10.1007/978-3-319-46726-9_46. 93

M. Rempfler, J. Lange, F. Jug, C. Blasse, E. W. Myers, B. H. Menze, and B. Andres. Efficient algorithms for moral lineage tracing. In *ICCV*, pages 4705–4714, 2017. doi: 10.1109/ICCV.2017.503. 4, 98, 99, 100

C. Rother, V. Kolmogorov, V. S. Lempitsky, and M. Szummer. Optimizing binary MRFs via extended roof duality. In *CVPR*, 2007. 24, 40, 41, 42

B. Rothfarb and I. T. Frisch. On the 3-commodity flow problem. *SIAM Journal on Applied Mathematics*, 17(1):46–58, 1969. 45

M. Sakarovitch. *The multi-commodity maximum flow problem.* PhD thesis, Operations Research Center, University of California, Berkeley, 1966. 45

T. Schank and D. Wagner. Finding, counting and listing all triangles in large graphs, an experimental study. In *Experimental and Efficient Algorithms, 4th International Workshop, (WEA), Proceedings*, pages 606–609, 2005. doi: 10.1007/11427186_54. 35

M. Schiegg, P. Hanslovsky, B. X. Kausler, L. Hufnagel, and F. A. Hamprecht. Conservation tracking. In *ICCV*, pages 2928–2935, 2013. 93

M. Schiegg, P. Hanslovsky, C. Haubold, U. Koethe, L. Hufnagel, and F. A. Hamprecht. Graphical model for joint segmentation and tracking of multiple dividing cells. *Bioinformatics*, 31:948–956, 2015. 93

A. Schrijver. *Theory of Linear and Integer Programming.* John Wiley & Sons, Inc., New York, NY, USA, 1986. ISBN 0-471-90854-1. 89

A. Schrijver. A short proof of guenin's characterization of weakly bipartite graphs. *J. Comb. Theory, Ser. B*, 85(2):255–260, 2002. doi: 10.1006/jctb.2001.2101. 46

A. Schrijver. *Combinatorial Optimization.* Springer, 2003. 8, 27, 45

P. D. Seymour. The matroids with the max-flow min-cut property. *J. Comb. Theory, Ser. B*, 23 (2-3):189–222, 1977. doi: 10.1016/0095-8956(77)90031-4. 47

P. D. Seymour. On lehman's width-length characterization. In *Polyhedral Combinatorics, Proceedings of a DIMACS Workshop, Morristown, New Jersey, USA, June 12-16, 1989*, pages 107–118, 1990. 46, 47

A. Shekhovtsov. *Exact and Partial Energy Minimization in Computer Vision.* PhD Thesis CTU–CMP–2013–24, Center for Machine Perception, K13133 FEE Czech Technical University in Prague, Prague, Czech Republic, 2013. 24, 25

A. Shekhovtsov. Maximum persistency in energy minimization. In *CVPR*, 2014. doi: 10.1109/CVPR.2014.152. 24, 25

A. Shekhovtsov. Higher order maximum persistency and comparison theorems. *Computer vision and image understanding*, 143(C):54–79, 2 2016. doi: 10.1016/j.cviu.2015.05.002. 24

A. Shekhovtsov, P. Swoboda, and B. Savchynskyy. Maximum persistency via iterative relaxed inference with graphical models. *TPAMI*, PP(99), 2017. doi: 10.1109/TPAMI.2017.2730884. 24

P. Swoboda and B. Andres. A message passing algorithm for the minimum cost multicut problem. In *CVPR*, 2017. doi: 10.1109/CVPR.2017.530. 11, 17, 19, 21, 22

P. Swoboda, B. Savchynskyy, J. H. Kappes, and C. Schnörr. Partial optimality via iterative pruning for the Potts model. In *Scale Space and Variational Methods (SSVM)*, 2013. 24

P. Swoboda, A. Shekhovtsov, J. H. Kappes, C. Schnörr, and B. Savchynskyy. Partial optimality by pruning for MAP-inference with general graphical models. *TPAMI*, 38(7):1370–1382, 7 2016. doi: 10.1109/TPAMI.2015.2484327. 24

D. Tang. Bi-path networks and multicommodity flows. *IEEE Transactions on Circuit Theory*, 11(4):468–473, December 1964. doi: 10.1109/TCT.1964.1082356. 45

S. Tang, B. Andres, M. Andriluka, and B. Schiele. Subgraph decomposition for multi-target tracking. In *CVPR*, pages 5033–5041, 2015. doi: 10.1109/CVPR.2015.7299138. 12, 93

S. Tang, M. Andriluka, B. Andres, and B. Schiele. Multiple people tracking by lifted multicut and person re-identification. In *CVPR*, 2017. doi: 10.1109/CVPR.2017.394. 12, 17, 59, 93

R. Tomer, K. Khairy, F. Amat, and P. J. Keller. Quantitative high-speed imaging of entire developing embryos with simultaneous multiview light-sheet microscopy. *Nature methods*, 9(7):755–763, 2012. 93

E. Türetken, G. González, C. Blum, and P. Fua. Automated reconstruction of dendritic and axonal trees by global optimization with geometric priors. *Neuroinformatics*, 9(2-3): 279–302, 2011. 93

E. Türetken, F. Benmansour, B. Andres, P. Glowacki, H. Pfister, and P. Fua. Reconstructing curvilinear networks using path classifiers and integer programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(12):2515–2530, 2016. 93

N. Veldt, A. I. Wirth, and D. F. Gleich. Correlation clustering with low-rank matrices. In *World Wide Web Conference (WWW)*, pages 1025–1034, 2017. doi: 10.1145/3038912.3052586. 10

N. Veldt, D. F. Gleich, and A. Wirth. A correlation clustering framework for community detection. In *World Wide Web Conference (WWW)*, pages 439–448, 2018. doi: 10.1145/3178876.3186110. 12, 17

T. Verma and D. Batra. Maxflow revisited: An empirical comparison of maxflow algorithms for dense vision problems. In *BMVC*, 2012. doi: 10.5244/C.26.61. 40

C. Wang and R. Zabih. Relaxation-based preprocessing techniques for Markov Random Field inference. In *CVPR*, 2016. doi: 10.1109/CVPR.2016.628. 24

D. Wang and R. Kleinberg. Analyzing quadratic unconstrained binary optimization problems via multicommodity flows. *Discrete Applied Mathematics*, 157(18):3746 – 3753, 2009. doi: https://doi.org/10.1016/j.dam.2009.07.009. 24

X. Wang, E. Türetken, F. Fleuret, and P. Fua. Tracking interacting objects optimally using integer programming. In *ECCV*, pages 17–32. Springer, 2014. 93

A. Weller. Characterizing tightness of LP relaxations by forbidding signed minors. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2016. 45

J. Yarkony, A. T. Ihler, and C. C. Fowlkes. Fast planar correlation clustering for image segmentation. In *ECCV 2012*, 2012. doi: 10.1007/978-3-642-33783-3_41. 11, 17

J. Yarkony, T. Beier, P. Baldi, and F. A. Hamprecht. Parallel multicut segmentation via dual decomposition. In A. Appice, M. Ceci, C. Loglisci, G. Manco, E. Masciari, and Z. W. Ras, editors, *New Frontiers in Mining Complex Patterns*, pages 56–68, Cham, 2015. Springer International Publishing. doi: 10.1007/978-3-319-17876-9_4. 10, 11, 17

I. K. Zervantonakis, S. K. Hughes-Alford, J. L. Charest, J. S. Condeelis, F. B. Gertler, and R. D. Kamm. Three-dimensional microfluidic model for tumor cell intravasation and endothelial barrier function. *Proceedings of the National Academy of Sciences*, 109(34): 13515–13520, 2012. 93