Matchings and Flows in Hypergraphs

Dissertation

zur Erlangung des akademischen Grades
doctor rerum naturalium (Dr. rer. nat.)

am Fachbereich Mathematik und Informatik
der Freien Universität Berlin

vorgelegt von

Isabel Beckenbach

Berlin, 2018

Erstgutachter:	Prof. Dr. Ralf Borndörfer Fachbereich Mathematik und Informatik Freie Universität Berlin
Zweitgutachter:	Prof. Dr. Winfried Hochstättler Fachbereich Mathematik FernUniversität Hagen
Tag der Disputation:	22. März 2019

Selbstständigkeitserklärung

Gemäß §7 Absatz 4 der Promotionsordnung des Fachbereichs Mathematik und Informatik der Freien Universität Berlin versichere ich, dass ich die vorliegende Arbeit selbstständig verfasst habe und alle verwendeten Hilfen und Hilfsmittel angegeben habe. Diese Arbeit wurde nicht schon einmal in einem früheren Promotionsverfahren eingereicht.

Berlin, den

Isabel Beckenbach

Preface

The well known concept of a graph is general enough to model and formulate a lot of mathematical and real world problems and concrete enough to allow for a rich structural and computational theory. Can one extend this theory to a larger class of mathematical structures? Which are the minimal requirements needed in order to generalize known graph theoretical results? Can one identify the true essence of a proof in this way? These were some of the questions that motivated the work on this thesis. The mathematical structures generalizing graphs considered in this scope are different classes of hypergraphs. The theoretical results that we generalize come from matching and flow theory.

This thesis consists of four major parts. In the first three we deal with matchings and generalizations of them in undirected hypergraphs and in the last part we develop a flow theory for directed hypergraphs. Table I gives a schematic overview of the problems we investigate in this thesis. The first two columns state which graph theoretic result we generalize to hypergraphs, in the third column the corresponding chapter or section of this thesis is cited, and the last column contains the classes of hypergraphs under consideration.

After a short introduction, we investigate in Chapter 2 matchings in hypergraphs generalizing bipartite graphs. Various generalizations of bipartite graphs to classes of "bipartite" hypergraphs exist in the literature. Among others we look at normal hypergraphs and give a new generalization of Hall's Theorem to this class of hypergraphs. Our main contribution in this part is to clarify the relationship between Kőnig's and Hall's Theorem for graphs and hypergraphs.

When working on matchings a natural next step is to investigate f-matchings and f-factors. This is done in Chapter 3, where we give existence theorems for perfect f-matchings and f-factors in various classes of hypergraphs generalizing bipartite graphs. These theorems were already known for so-called unimodular hypergraphs, which are hypergraphs corresponding to totally unimodular binary matrices. Theorems concerning these matrices are often proven using linear programming methods. Here, we use purely combinatorial methods revealing the combinatorial nature of these problems and giving new methods in the field of totally unimodular matrices. For the larger class of Mengerian hypergraphs the proven results are new to the best of our knowledge. It is again possible to use linear programming methods, however, we see the pure combinatorial proofs as a main contribution.

In Chapter 2 and Chapter 3 we investigate hypergraphs with some additional properties in order to characterize the existence of some substructure (perfect matching in the first case, perfect f-matchings and f-factors in the second). Another interesting possibility is to look at the structure of hypergraphs having a perfect matching. Graphs with a perfect matching can be decomposed along special cuts into so-called bricks and braces- the indecomposable elements. The way one decomposes a graph is not unique. However, in the end, one gets the same list of bricks and braces. In Chapter 4 we define a tight cut decomposition for hypergraphs and show that it is unique for uniform hypergraphs, which are hypergraphs in which all hyperedges have the same size. Furthermore, we investigate hypergraphs in which every matching of size k lies in some perfect matching for some natural number k. A hypergraph with this property is called k-extendable. Here, we both consider the case of general hypergraphs and that of balanced hypergraphs. In balanced uniform hypergraphs we characterize k-extendability generalizing known results for bipartite graphs.

Our last chapter deals with flows in directed hypergraphs and is inspired by an application in the field of rolling stock rotation planning. In this application it is more convenient to work with directed hypergraphs rather than directed graphs. In order to model coupling activities and other side constraints hyperarcs are added, where a hyperarc consists of a set of arcs having no common tail or head vertices. Some results on flows in directed graphs can be transferred to flows on these specific directed hypergraphs, e.g., the concept of a residual network or the network simplex algorithm. On the other hand, some properties are lost as for example the integrality of a flow for integral input data and the equivalence of the path-based and the arc-based linear programming formulation of the maximum s, t-flow or minimum cost flow problem.

All in all, we show that it is worthwhile to investigate hypergraphic generalizations of graph problems, and that it is possible to obtain nice structural results and combinatorial algorithms. This thesis is a first step towards further research on algorithmic and structural hypergraph theory.

	Graph Results	Where	Hypergraph Classes
Bipartite graphs		Chapter 2 & 3	
	Existence of perfect matchings	Sec. 2.3.2	perfect, balanced, normal hypergraphs
	Existence of (g, f)-factors,	Sec. 3.2	unimodular hypergraphs
	perfect f-matchings,	Sec. 3.3.1	Mengerian uniform hypergraphs
	f-factors	Sec. 3.3.2	balanced uniform hypergraphs
Matching covered graphs		Chapter 4	
	Extendability	Sec. 4.2	general and balanced hypergraphs
	Tight Cuts	Sec.4.3	general hypergraphs
	Tight Cut Decomposition	Sec. 4.4	uniform hypergraphs
Directed graphs		Chapter 5	
	Maximum Flow and Minimum Cost Flow	Sec. 5.2	directed hypergraphs
	Network Simplex Algorithm	Sec. 5.3	graph-based directed hypergraphs
	for the minimum cost flow problem		

Table I: Contents of this thesis.

Publications and Collaborations

Most of the new results of this thesis have been published in or have been submitted to peer-reviewed conference proceedings or journals. Moreover, a lot of work has been done in collaboration with colleagues from Berlin and Aachen.

- Theorem 2.13 in Chapter 2 is joint work with Ralf Borndörfer and appeared in [Beckenbach and Borndörfer, 2016]. All other results of Section 2.2 have not been published yet and are independent work of the author of this thesis.
- The results of Section 2.3 are joint work with Ralf Borndörfer and are published in [Beckenbach and Borndörfer, 2018].
- Section 3.2 in Chapter 3 is joint work with Britta Peis, Oliver Schaudt, and Robert Scheidweiler. A preprint version [Beckenbach et al., 2017] can be found on the preprint server `https://opus4.kobv.de/opus4-zib/home` of the Zuse Institute Berlin. The article is currently under review.
- The results of Section 3.3 in Chapter 3 are joint work with Robert Scheidweiler and are published in [Beckenbach and Scheidweiler, 2017].
- Section 4.3, 4.4 and Subsection 4.2.2 in Chapter 4 are joint work with Meike Hatzel, and Sebastian Wiederrecht.

Funding

From November 2013 to August 2014 I was supported by a scholarship from Zuse Institute Berlin. Since September 2014 the research for this thesis was conducted within the project MODAL-RailLab, which is funded by the German Federal Ministry of Education and Research (BMBF) and DB Fernverkehr AG.

Acknowledgements

This thesis would not have been possible without the help and support of a lot of people:

First of all, my parents supported me a lot and ensured that I could focus on my studies the whole time.

Then, I would like to thank my supervisor Ralf Borndörfer for giving me the chance to write my PhD Thesis at Zuse Institute Berlin. He always gave me a lot of freedom to pursue my own research interests but also took a lot of his little free time to listen to my ideas.

I enjoyed the nice working atmosphere at Zuse Institute Berlin with all the BBQ's, cakes, Game Nights, and so on. I want to thank all my colleagues from the optimization department. Especially, I thank Boris, Stanley, and Niels for proof-reading parts of this thesis.

I was really lucky to meet Robert Scheidweiler at the Operations Research Conference in Aachen 2014. This was the first time that I met someone else who was interested in my work. I would like to thank him for the inspiring discussions and his invitation to Aachen. I would also like to thank my other co-authors from Aachen, Britta Peis and Oliver Schaudt. Unfortunately, we never met in person to discuss our ideas.

By the time Robert left the academic world a former student of his, Sebastian Wiederrecht, wrote me an e-mail asking whether I was interested in a joint project with his colleague Meike Hatzel on the structure of hypergraphs with a perfect matching. This was the beginning of an inspiring work that lead to the uniqueness result of the tight cut decomposition in uniform hypergraphs presented in Section 4. I really enjoyed our meetings and they motivated me a lot. Meike also read large parts of this thesis very carefully and helped me a lot correcting stupid mistakes. Thank you very much!

Moreover, I am grateful to Winfried Hochstättler for examining this dissertation and for giving me useful tips during the last years.

Last but not least I would like to thank Niels and Felix for all the love and joy they bring into my life.

Contents

List of Figures

List of Tables

Chapter 1

An Introduction to Hypergraphs

Usually a graph is seen as an abstract structure modeling pairwise connections of a set of objects called vertices. Two vertices can be either connected by an edge or be independent from each other. One way to generalize this concept is to allow an edge to connect an arbitrary number of vertices. Such edges are called hyperedges and are nothing other than subsets of the vertex set. A set of vertices together with a family of hyperedges forms a hypergraph. Hypergraphs are very abstract objects with less structure than graphs. However, in some applications it is more appropriate to use hypergraphs instead of graphs as a modeling paradigm. For example, in the field of transportation planning, hypergraphs are used to model tram lines in [Karbstein, 2013]) or coupling of railway vehicles in [Borndörfer et al., 2012]. Hypergraphs also occur in the field of logic and artificial intelligence hypergraphs, see [Eiter and Gottlob, 1995]. Furthermore, both directed and undirected hypergraphs are successfully used in the field of biological networks analysis, see [Klamt et al., 2009] for a short overview. For example, protein interactions involve often more than one protein, thus they can be modeled more accurately if hyperedges instead of edges are used.

As a drawback of the modeling power of hypergraphs, a lot of combinatorial optimization problems that are solvable in polynomial time on graphs are $\mathcal{NP}$-hard on hypergraphs. Furthermore, there is no nice structural theory available as in the graph case. Therefore, we will investigate hypergraphs with some additional structure. In Chapter 2 and 3 we look at hypergraphs generalizing bipartite graphs, in Chapter 4 at hypergraphs in which every hyperedge is contained in a perfect matching, and in Chapter 5 at hypergraphs based on directed graphs. Except of the last chapter all hypergraphs considered in this thesis are undirected. Thus, we postpone all definitions and notations for directed hypergraphs to Chapter 5 and concentrate on the undirected case for now.

This chapter gives an introduction into hypergraph theory to such an extend as it is needed in the following three chapters. For more details the reader is referred to [Berge, 1984]. We assume that the reader is familiar with basic graph theory and linear programming. After introducing some general notation, we give some elementary definitions concerning hypergraphs. In the second section, we give an overview about hypergraphs generalizing bipartite graphs.

General Notation

The sets of rational, integers, and natural numbers are denoted by $\mathbb{Q}, \mathbb{Z}$, and $\mathbb{N}$, respectively, where 0 is not treated as a natural number. The sets $\mathbb{Q}_{\geq 0}$, $\mathbb{Z}_{\geq 0}$ consist of all non-negative rationals, and all non-negative integers. For a natural number n we abbreviate the set $\{1, \dots, n\}$ by $[n]$.

The function $(\cdot)_+ : \mathbb{Q} \to \mathbb{Q}_{\geq 0}$ is defined by $(q)_+ = \begin{cases} q, & \text{if } q \geq 0 \\ 0, & \text{otherwise} \end{cases}$ for $q \in \mathbb{Q}$.

Given a finite set S we denote by K^S the set of all function from S to K. If $x \in K^S$, then we sometimes also view x as a vector indexed by S, and write x_s instead of $x(s)$ for $s \in S$. Furthermore, for any non-empty subset T of S the sum $\sum_{s \in T} x(s)$ is abbreviated by $x(T)$. If there is some partial order $\leq_K$ on K given, then $x \leq y$ for $x, y \in K^S$ if and only if $x(s) \leq y(s)$ for all $s \in S$. Finally, for some subset T of S and $x \in K^S$ we let $x_T \in K^T$ be the restriction of x to T.

If S and T are two finite sets we denote by $K^{S \times T}$ the set of all matrices with entries in K whose rows are indexed by S and whose columns by T.

1.1 Basic Hypergraph Definitions

The literature on hypergraphs is not very consistent and there are various different notations and concepts. To avoid confusion, we start with a short section summarizing the basic notions needed in the remainder of the thesis.

Definition 1.1 (Hypergraph)**.** A *hypergraph* H is a pair $(V(H), E(H)$, where $V(H)$ is a finite set and $E(H)$ is a finite family of non-empty subsets of $V(H)$. An element $v \in V(H)$ is called a *vertex*, and $e \in E(H)$ a *hyperedge* of H. Given a vertex v and a hyperedge e, we say that v and e are *incident* if $v \in e$. The family of all hyperedges incident to a vertex is denoted by $\delta_H(v)$, and its size is the *degree* of vertex v, denoted by $\deg_H(v)$.

Our definition of a hypergraph allows that two distinct hyperedges contain the same set of vertices. Such hyperedges are called *parallel*. A *simple hypergraph* is a hypergraph without parallel hyperedges. It is possible that a vertex is not contained in any hyperedge, in which case we call the vertex *isolated*. Isolated vertices are not allowed in the definition of a hypergraph due in [Berge, 1984]. In contrast to Berge, we use the operator notation[1] for the vertex and hyperedge set of a hypergraph. This means that the vertex set of a hypergraph H is denoted by $V(H)$ and the hyperedge set by $E(H)$.

When we draw a hypergraph we represent vertices as points and hyperedges as closed simple curves enclosing the vertices they contain. Figure 1.1 shows an ex-

[1] see point 35 in `https://faculty.math.illinois.edu/~west/grammar.html`

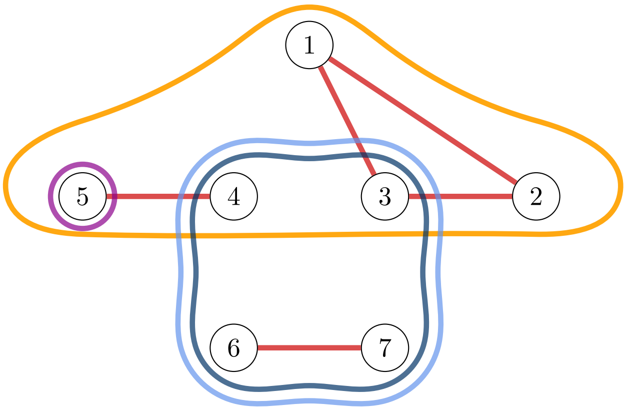


Figure 1.1: Drawing of a hypergraph.

ample of a drawing of a hypergraph. The yellow curve represents the hyperedge containing the vertices $1, 2, 3, 4, 5$, there are two parallel hyperedges containing the vertices $3, 4, 6, 7$ (dark blue and light blue), there is a hyperedge of size one containing vertex 5, and five hyperedges of size two. For better readability we draw edges (hyperedges of size two) just as line segments connecting its two end vertices, and we might use different colors for hyperedges. The tikz code for drawing an undirected hyperedge was adapted from that of Meike Hatzel using the hobby package. [2]

In general, we want to consider hypergraphs only up to relabeling of vertices and hyperedges, i.e., up to isomorphism. As we deal with hypergraphs having parallel hyperedges, we have to be careful in the definition of a hypergraph isomorphism. Namely, we do not only need a bijection between the vertex sets of two hypergraphs but also between the hyperedge families, see [Berge, 1975].

Definition 1.2 (Hypergraph isomorphism)**.** Two hypergraphs H_1 and H_2 are isomorphic if there exist bijections $f : V(H_1) \to V(H_2)$ and $g : E(H_1) \to E(H_2)$ such that $\{f(v) : v \in e\} = g(e)$ for all $e \in E(H_1)$.

There is a correspondence between isomorphism classes of hypergraphs and binary matrices up to row and column permutations. Namely, given a hypergraph H we define a binary $|V(H)| \times |E(H)|$ matrix $A(H)$ with rows indexed by the vertices and columns by the hyperedges of H by

$$A(H)_{v,e} := \begin{cases} 1, & \text{if } v \in e \\ 0, & \text{otherwise.} \end{cases}$$

The matrix $A(H)$ is the *incidence matrix* of H. On the other hand, if $A \in \{0, 1\}^{n \times m}$ is a binary matrix, then its associated hypergraph $H(A)$ has vertex set $[n]$ and

[2] https://ctan.org/pkg/hobby

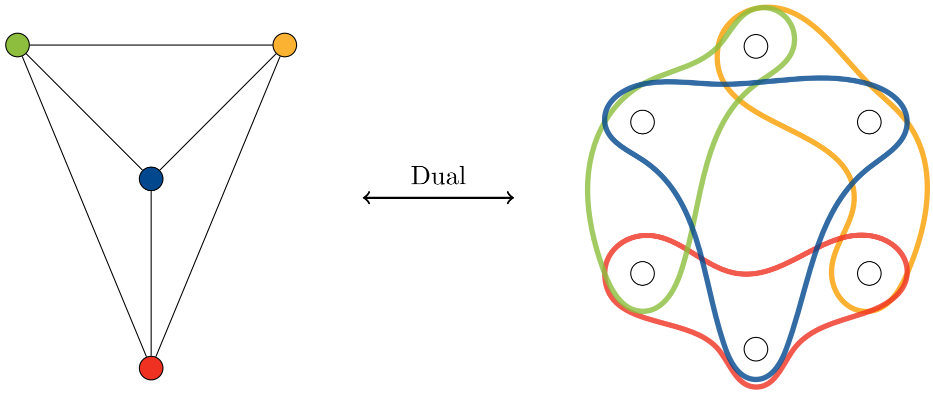


Figure 1.2: K_4 and its dual hypergraph $(K_4)^*$.

hyperedges $e_j := \{i : A_{i,j} = 1\}$ for $j \in [m]$. This means that every column of A induces a hyperedge of $H(A)$. With this definition the incidence matrix of $H(A)$ is A for every binary matrix A, and every hypergraph H is isomorphic to $H(A(H))$. Therefore, binary matrices and hypergraphs are essentially the same. We work with hypergraphs as we want to highlight the combinatorial nature of these objects, and use graph theoretic ideas.

An important concept is that of the dual of a hypergraph for which we switch the roles of vertices and hyperedges, see Figure 1.2 for an example.

Definition 1.3 (Dual hypergraph)**.** Let H be a hypergraph. For every $e \in E(H)$ let u_e be a new vertex not in $V(H)$, and for every vertex $v \in V(H)$ we define a hyperedge e_v by $e_v := \{u_e : e \in E(H), v \in e\}$. The *dual hypergraph* H^* of H has $\{u_e : e \in E(H)\}$ as its vertex set, and its hyperedge family consists of all hyperedges e_v for $v \in V(H)$.

Taking the dual of a hypergraph corresponds to matrix transposition, namely $A(H^*) = A(H)^T$.

A hypergraph can be represented by a graph in various ways. Mainly two possibilities are considered in the literature. In the first one, the incidence structure of the vertices and hyperedges is represented by a bipartite graph.

Definition 1.4 (Bipartite Representation)**.** The *bipartite representation* of a hypergraph H is the bipartite graph with vertex set $V(H) \cup E(H)$ and edges $\{v, e\}$ for every $v \in V(H), e \in E(H)$ with $v \in e$. It is denoted by $\mathrm{Bip}(H)$.

Figure 1.3 shows the bipartite representation of the hypergraph drawn in Figure 1.1. The bipartite representation captures a lot of information of a hypergraph. Namely, $\mathrm{Bip}(H_1)$ is isomorphic to $\mathrm{Bip}(H_2)$ for two hypergraphs H_1, H_2 if and only if H_1 is isomorphic to H_2 or H_2^*.

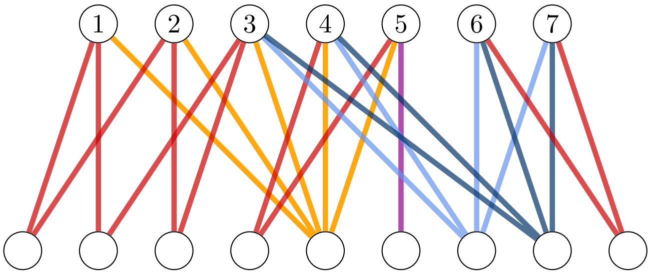


Figure 1.3: The bipartite representation of the hypergraph depicted in Figure 1.1.

A second way to represent a hypergraph by a graph is to model the pairwise intersection of the hyperedges by edges of a graph.

Definition 1.5 (Line Graph)**.** The *line graph* of a hypergraph H has a vertex v_e for every $e \in E(H)$ and edges $\{v_e, v_f\}$ for every $e, f \in E(H)$ with $e \cap f \neq \emptyset$, $v_e \neq v_f$. It is denoted by $L(H)$.

In contrast to the bipartite representation, the line graph of a hypergraph and that of its dual are in general not isomorphic and two non-isomorphic hypergraphs might have isomorphic line graphs. For example, we allow a hypergraph to have isolated vertices, which are vertices not contained in any hyperedge, and the existence of such vertices cannot be seen on the line graph. Even if we only consider hypergraphs without isolated vertices, it is possible that the line graphs of two non-isomorphic hypergraphs are isomorphic. An example is depicted in Figure 1.4. The hypergraph on the left and that on the right both have four hyperedges that intersect pairwise. This implies that the line graph of both is the complete graph on four vertices. As the number of vertices differs, the two hypergraphs cannot be isomorphic.

We continue with the definition of some basic hypergraph parameters. As in the graph case, the *maximum degree* $\Delta(H)$ of a hypergraph H is the maximum of the degrees of its vertices. A *d-regular hypergraph* is a hypergraph in which all vertices have degree d, and a *regular hypergraph* is a hypergraph that is d-regular for some $d \in \mathbb{N}$. The hyperedges of a hypergraph can vary in size. The *rank* of H, denoted by $r(H)$, is equal to the maximum size of a hyperedge of H. The minimum size of a hyperedge is abbreviated by $s(H)$ and called the *lower rank* of H. A *uniform hypergraph* is a hypergraph in which all hyperedges have the same size or equivalently the rank of the hypergraph is equal to its lower rank. If additionally the size r of all hyperedges is known, then the hypergraph is called *r-uniform*. A 2-uniform hypergraph is just a graph without loops. If the vertex set of an r-uniform hypergraph H can be partitioned into r disjoint subsets $V_1, \ldots, V_r$ such that each hyperedge intersects each of them in exactly one vertex, then H is *r-partite*, and

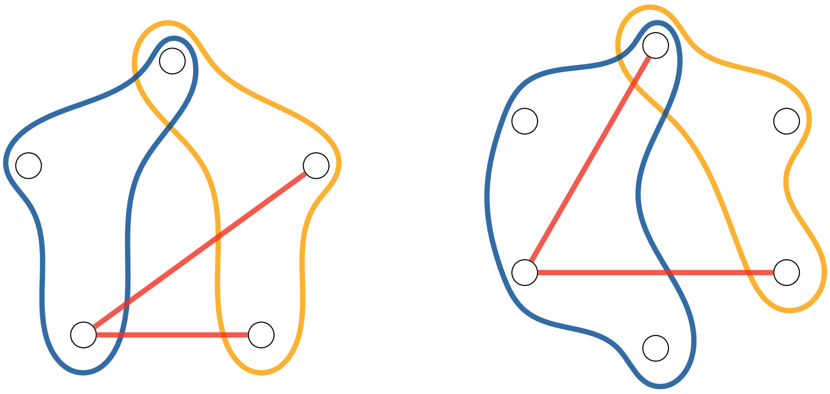
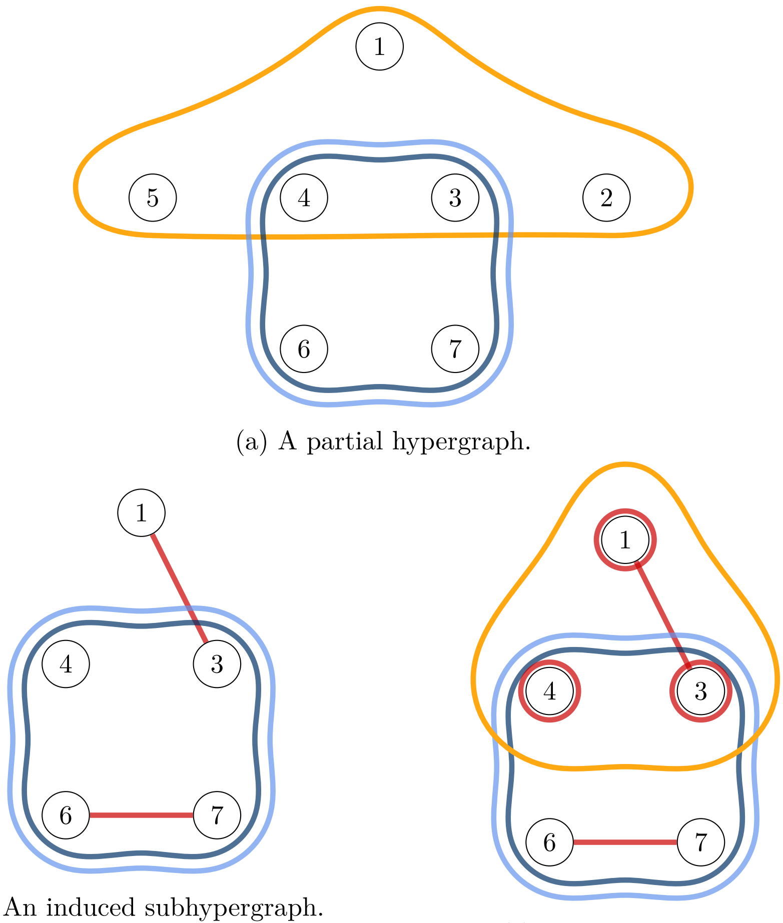


Figure 1.4: Two non-isomorphic hypergraphs that both have a line graph isomorphic to K_4.

$V_1, \dots, V_r$ is an *r-partition* of H.

In contrast to the graph case, there are various different definitions of a "subhypergraph". Given a hypergraph H, we look at three possibilities to restrict H to some subset of hyperedges and vertices. First, given a non-empty subfamily F of the hyperedges of H, we restrict $E(H)$ to F and do not change the vertex set. The resulting hypergraph is denoted by $H[F]$, and every hypergraph obtained from H in this way is called a *partial hypergraph* of H. When restricting H to some subset S of vertices we look at two possibilities to restrict the hyperedges. If we only consider hyperedges lying completely in S, then the resulting hypergraph is called the *subhypergraph induced by* S and denoted by $H[S]$. On 2-uniform hypergraphs this definition conforms with the usual one of an induced subgraph. Another possibility to restrict $E(H)$ is to intersect each single hyperedge with S and keep the hyperedges that remain non-empty. Formally, the *subhypergraph restricted to* S has vertex set S and contains all hyperedges of the form $e \cap S$, where $e \in E(H)$ has a non-empty intersection with S. It is denoted by $H(S)$. In the remainder of this thesis, a *subhypergraph* of H is a subhypergraph restricted to some set $S \subseteq V(H)$ and an *induced subhypergraph* is a subhypergraph induced by some set $S \subseteq V(H)$. Furthermore, we denote by $H - S$ the subhypergraph induced by $V(H) \setminus S$, and by $H \setminus S$ the subhypergraph restricted to $V(H) \setminus S$, i.e., $H - S = H[V(H) \setminus S]$ and $H \setminus S = H(V(H) \setminus S)$. Finally, a *partial subhypergraph* of H is a partial hypergraph of a subhypergraph of H.

We demonstrate the different definitions of "subhypergraph" on the hypergraph H depicted in Figure 1.1. We start with the most intuitive of the three notions: the partial hypergraph. In this case, we only restrict the family of hyperedges to some subfamily and leave the vertex set unchanged. For example, Figure 1.5a shows



(a) A partial hypergraph.

(b) An induced subhypergraph.

(c) A subhypergraph.

Figure 1.5: The three different notions of "subhypergraph".

the partial hypergraph obtained from H by restricting the hyperedges to those of size greater than two. To illustrate the two other notions of a "subhypergraph" we consider the set $S = \{1, 3, 4, 6, 7\}$. The subhypergraph $H[S]$ induced by S contains only those hyperedges lying completely in S, which are in this case the two parallel hyperedges containing $3, 4, 6, 7$ and the edges $\{1, 3\}, \{6, 7\}$. On the other hand, the subhypergraph $H(S)$ restricted to S contains the intersection of all hyperedges with S if it is non-empty. In our concrete example this means that $H(S)$ contains all four hyperedges of $H[S]$ and additionally the hyperedges $\{1\}, \{3\}, \{4\}$, and $\{1, 3, 4\}$.

Now, we come to one of the most important definitions in this thesis; the one of a matching and its different notions of maximality. The definition of a matching in graphs carries over to hypergraphs directly. However, in contrast to graphs it is not clear when a matching is maximum. Should a matching cover as much vertices as possible or should it contain as many hyperedges as possible? In a non-uniform hypergraph both approaches can lead to different results. For example, the hypergraph depicted in Figure 1.1 has a matching of size two covering all seven hyperedges, which consists of the hyperedges $\{1, 2, 3, 4, 5\}$ and $\{6, 7\}$. However, it has also a matching of size three, namely, $\{\{1, 2\}, \{4, 5\}, \{6, 7\}\}$, which covers only six vertices. Therefore, we consider both variants, where we use the notions introduced in [Scheidweiler, 2011].

Definition 1.6 (Matching)**.** A *matching* M in a hypergraph H is a set of hyperedges of H such that every vertex is contained in at most one of the hyperedges of M. If a vertex is contained in a hyperedge of M, then it is *covered* by M, and otherwise *exposed* by M. The set of all vertices covered by M is denoted by $V(M)$. A *perfect matching* is a matching covering all vertices.

Furthermore, a matching M is called *V-maximum* if no matching covering more vertices than M exists, and it is called *E-maximum* if there exists no matching of larger cardinality. The size of a V-maximum matching is denoted by $\nu_V(H)$ and that of an E-maximum matching by $\nu_E(H)$. In general, given any function $b : E(H) \to \mathbb{Q}$ the maximum of $\sum_{e\in M} b(e)$ over all matchings M in H is denoted by $\nu_b(H)$.

We again consider the hypergraph of Figure 1.1 to illustrate the previous definitions. The hyperedges $\{1, 2, 3, 4, 5\}$ and $\{6, 7\}$ form a V-maximum matching that is also perfect. An E-maximum matching is for example given by the edges $\{1, 2\}$, $\{6, 7\}$, and $\{4, 5\}$. This matching is not perfect because it exposes vertex 3.

The dual concept of matchings are vertex covers, which we directly define for the weighted case.

Definition 1.7 (Vertex cover)**.** Given a hypergraph H and a function $b : E(H) \to \mathbb{Z}$ on the hyperedges of H, a *b-vertex cover* is a function $x : V(H) \to \mathbb{Z}_{\geq 0}$ such that $\sum_{v\in e} x(v) \geq b(e)$ holds for all hyperedges $e \in E(H)$. The *size* of a vertex cover x is $\sum_{v\in V(H)} x(v)$, and a b-vertex cover is *minimum* if it is of minimum size. The size of minimum b-vertex cover of H is denoted by $\tau_b(H)$.

If $b(e) = 1$ for all $e \in E(H)$ a b-vertex cover is called an *E-vertex cover*, and the minimum size of an E-vertex cover is denoted by $\tau_E(H)$. If $b(e) = |e|$ for all $e \in E(H)$ a b-vertex is called a *V-vertex cover* and $\tau_V(H)$ is the minimum size of a V-vertex cover.

For a given function b on the hyperedges of a hypergraph H, every minimum b-vertex cover satisfies $x(v) \leq \max_{e\in E(H)} b(e)$ for all $v \in V(H)$. Otherwise, if $x(w) \geq \max_{e\in E(H)} b(e) + 1$ for some $w \in V(H)$, then $x' : V(H) \to \mathbb{Q}_{\geq 0}$ defined by $x'(w) = x(w) - 1$ and $x'(v) = x(v)$ for all $v \in V(H)\setminus\{w\}$ is a b-vertex cover of size less than x, contradicting that x is of minimum size. Thus, $x(v) \leq \max_{e \in E(H)} b(e)$ for all $v \in V(H)$. In particular, every minimum E-vertex cover has range in $\{0, 1\}$, and can therefore be identified by a subset of the vertices that intersects each hyperedge in at least one vertex.

In our example of Figure 1.1 the set $\{1, 2, 5, 6\}$ is a minimum E-vertex cover, and the function that assigns 1 to every vertex is a minimum V-vertex cover.

Every minimum E-vertex cover has to contain at least one vertex from each hyperedge of a maximum E-matching, thus $\tau_E(H) \geq \nu_E(H)$ follows. Also the inequality $\tau_V(H) \geq \nu_V(H)$ holds. Namely, if x is a minimum V-vertex cover and M is a maximum V-matching, then

$$\sum_{v \in V(H)} x(v) \geq \sum_{v \in V(M)} x(v) = \sum_{e \in M} \sum_{v \in e} x(v) \geq \sum_{e \in M} |e| = |V(M)|.$$

It is possible that none, exactly one, or both of the inequalities $\tau_E(H) \geq \nu_E(H)$, $\tau_V(H) \geq \nu_V(H)$ are strict. In our running example we have $\tau_V(H) = 7 = \nu_V(H)$ and $\tau_E(H) = 4 > 3 = \nu_E(H)$. Indeed, every hypergraph with a perfect matching satisfies $\nu_V(H) = |V(H)| = \tau_V(H)$ but not necessarily $\nu_E(H) = \tau_E(H)$.

Now, we define two other concepts related to matchings and vertex covers. First, we look at sets of hyperedges covering all vertices.

Definition 1.8 (Hyperedge cover)**.** A *hyperedge cover* C is a set of hyperedges of H such that every vertex is contained in at least one hyperedge of C.

A lower bound on the minimum size of an edge cover in a graph is given by the maximum size of a stable set. In bipartite graphs both values are equal. We define stable sets for hypergraphs as follows:

Definition 1.9 (Stable set)**.** A subset S of the vertex set of a hypergraph H is *stable* if $|e \cap S| \leq 1$ for all $e \in E(H)$.

As in the graph case, the maximum size of a stable set is at most the minimum size of a hyperedge cover because a stable set contains at most one vertex from each hyperedge of a hyperedge cover.

In Chapter 3 we look at a generalization of matchings, where for every vertex we are given a lower and upper bound on the number of times this vertex should be covered.

Definition 1.10 ((g, f)-matching, (g, f)-factor)**.** Given a hypergraph H, and two functions $f, g : V(H) \to \mathbb{Z}_{\geq 0}$, a function $x : E(H) \to \mathbb{Z}_{\geq 0}$ is called a (g, f)-*matching* if $g(v) \leq x(\delta_H(v)) \leq f(v)$ holds for all $v \in V(H)$. If an additional function $c : E(H) \to \mathbb{Z}_{\geq 0}$ is given, then a *c-capacitated* (g, f)*-matching* is a (g, f)-matching x with $x(e) \leq c(e)$ for all $e \in E(H)$.

In the special case that $c(e) = 1$ for all $e \in E(H)$ we call a c-capacitated (g, f)-matching just a *(g, f)-factor*. Furthermore, an *f-matching* is a (g, f)-matching for $g(v) = 0$ for all $v \in V(H)$, a *perfect f-matching* is an (f, f)-matching, and an *f-factor* is an (f, f)-factor.

We use the term factor instead of matching if we are only allowed to take each hyperedge once, i.e., a factor can be seen as a set of hyperedges.

The definition of (g, f)-matchings indeed generalizes matchings, hyperedge covers, and perfect matchings: Namely, if f is constantly one, and h constantly equal to the maximum degree $\Delta(H)$ of H, then an f-matching is just a matching, an (f, h)-factor is a hyperedge cover, and a perfect f-matching is a perfect matching.

The concept of a path or a cycle can be defined similarly as in the graph case, except that it is possible that a hyperedge connects more than two vertices of a path or cycle, in which case we can shorten it.

Definition 1.11 (Paths and cycles)**.** A *path P* in a hypergraph H is an alternating sequence $(v_1, e_1, v_2, e_2, \ldots, v_{k-1}, e_{k-1}, v_k)$ of vertices and hyperedges such that $v_i, v_{i+1} \in e_i$ for $i \in [k]$. If all vertices and all hyperedges of P are distinct, then the path P is called *simple.* If $v_1 = v_k$, and otherwise all vertices are distinct, then P is called a *cycle.* The *length* of a path or cycle is its number of not necessarily distinct hyperedges. An *odd cycle* is a cycle of odd length, and an *even cycle* a cycle of even length.

A path or cycle $(v_1, e_1, \ldots, v_{k-1}, e_{k-1}, v_k)$ is called *strong* if $|e_i \cap \{v_1, \ldots, v_k\}| = 2$ for all $i \in [k-1]$.

Example 1.12. Figure 1.6 shows an odd cycle $(1, e_1, 2, e_2, 3, e_3, 4, e_4, 5, e_5, 1)$ in a hypergraph. This cycle is not strong because e_3 contains three vertices of it. It can be split into the two strong odd cycles $(1, e_1, 2, e_2, 3, e_3, 1)$ and $(1, e_3, 4, e_4, 5, e_5, 1)$.

A cycle of length k of a hypergraph corresponds to a cycle of length $2k$ in its bipartite representation, and vice versa. Using this correspondence, we observe that a cycle in a hypergraph is strong if and only if the corresponding cycle in the bipartite representation is induced, where a cycle $C = (v_1, e_1, \ldots, e_{k-1}, v_k, e_k, v_1)$ in a graph G is induced if G contains no edge $\{v_i, v_j\}$ for $i = k$, and $j \neq 1$, or $i \in [k-1]$ and $j \neq i + 1$.

As in the graph case, a hypergraph H is *connected* if for every pair of vertices $s, t \in V(H)$ there exists a path starting in s and ending in t. It is also possible to generalize k-edge and k-vertex connectivity to hypergraphs.

Definition 1.13 (k-connectivity)**.** Let $k \in \mathbb{N}$ be a natural number, H be a hypergraph, and $s, t \in V(H)$ be two distinct vertices. We say that s and t are *k-hyperedge connected* in H if H has more than k hyperedges and s and t are connected in $H[E \setminus X]$ for every set $X \subseteq E(H)$ of size $k - 1$. Furthermore, s and t are *k-vertex*

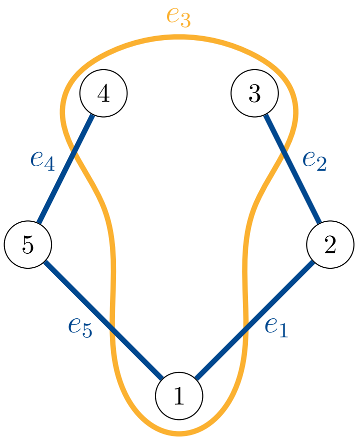


Figure 1.6: An odd cycle that is not strong.

connected in H if H has more than k vertices and s and t are connected in $H \setminus S$ for every set $S \subseteq V(H)$ of size $k-1$ with $s, t \notin S$.

In the graph case Menger's Theorem relates k-connectivity to cuts. This can be done in a similar way for hypergraphs. First, we define cuts in hypergraphs formally.

Definition 1.14 (Cut)**.** Given a hypergraph H and a subset S of its vertex set, the *cut induced by* S is the set of hyperedges of H that have a non-empty intersection with both S and $V(H) \setminus S$, and it is denoted by $\delta_H(S)$.

A set $C \subseteq E(H)$ of hyperedges is a *cut* if there exists a set of vertices $S \subseteq V(H)$ with $C = \delta_H(S)$. In this case, S and $V(H) \setminus S$ are called the shores of the cut C.

Given two distinct vertices $s, t \in V(H)$ an s,t-cut of H is a cut of the form $\delta_H(S)$ for some $S \subseteq V(H)$ with $s \in S$ and $t \notin S$.

The bipartite representation of a hypergraph can be used to prove a Menger-type theorem for hypergraphs, see [Frank, 2011].

Theorem 1.15 (Menger's Theorem for Hypergraphs)**.** *Let H be a hypergraph and s, t be two distinct vertices of H. The minimum cardinality of an s,t-cut is equal to the maximum number of hyperedge disjoint paths from s to t.*

Corollary 1.16. *Two distinct vertices s and t in a hypergraph with at least $k+1$ hyperedges are k-hyperedge connected if and only if there are k hyperedge disjoint paths connecting s and t.*

Via duality we can transform statements about hyperedge-connectivity into ones about vertex-connectivity. In particular, we prove a vertex version of Menger's

Theorem. We call two s, t-paths *internally vertex disjoint* if they only share the vertices s and t.

Corollary 1.17. *Let H be a hypergraph on at least $k+1$ vertices, $s, t \in V(H)$ be two distinct vertices such that no hyperedge $e \in E(H)$ exists containing both s and t. The vertices s, t are k-vertex connected in H if and only if there are k internally vertex disjoint paths in H connecting them.*

Proof. Suppose that $P_1, \ldots, P_k$ are k internally vertex disjoint s, t-paths but s, t are not k-vertex connected in H. Let S be a vertex set of size $k-1$ such that s and t are not connected in $H \setminus S$. The set S has to intersect each of the k paths in at least one vertex because s and t are not connected in $H \setminus S$. But this is not possible because $P_1, \ldots, P_k$ are internally vertex disjoint and S has size $k-1$.

On the other hand, suppose that s and t are k-vertex connected in H. The vertices s and t correspond to hyperedges e_s and e_t in the dual H^*. Let $\tilde{H}^*$ be the hypergraph obtained from H^* by adding two new vertices s^*, t^* and replacing e_s by k parallel hyperedges of the form $e_s \cup \{s^*\}$ and e_t by k parallel hyperedges $e_t \cup \{t^*\}$. Suppose that there exists an s^*, t^*-cut C of size less than k. In this case, C does not contain any of the k parallel hyperedges $e_s \cup \{s^*\}$ or $e_t \cup \{t^*\}$. Let $S \subseteq V(H)$ be the set of vertices in H corresponding to C. Every path from s to t uses a vertex $v \in S$ because such a path corresponds to a path from s^* to t^* and the paths from s^* to t^* have to intersect C. This means that $H \setminus S$ is disconnected. But S has size less than k and H is assumed to be k-connected. Thus, the minimum size of a cut separating s^* and t^* in $\tilde{H}^*$ is at least k. By Theorem 1.15, there exist k hyperedge disjoint paths $\tilde{P}_1, \ldots, \tilde{P}_k$ from s^* to t^* in $\tilde{H}^*$. As s and t are not contained in a common hyperedge of H, there exists no vertex incident to both of s^* and t^* in $\tilde{H}^*$. In particular, this implies that $\tilde{P}_1, \ldots, \tilde{P}_k$ contain at least three hyperedges and correspond to k distinct paths in H. These k paths are internally vertex disjoint s, t-paths in H. □

1.2 Hypergraphs Generalizing Bipartite Graphs

There exist several generalizations of the notion of "bipartiteness" to hypergraphs in the literature, each starting with a different characterization of bipartite graphs. Namely, a graph G is bipartite if and only if one of the following statements holds:

(a) its incidence matrix is totally unimodular (unimodular hypergraphs),

(b) it has no cycle of odd length (balanced hypergraphs),

(c) the edge set of every subgraph G' of G can be partitioned into $\Delta(G')$ matchings (normal hypergraphs),

(d) a weighted variant of Kőnig's Theorem holds (Mengerian hypergraphs),

(e) the vertex set can be partitioned into two disjoint subsets U and W such that $|e \cap U| = |e \cap W|$ holds for every edge e of G (partitioned hypergraphs).

The names in brackets indicate the hypergraphic generalization corresponding to the given statement. It turns out that the five statements above are not equivalent for hypergraphs anymore. In this section we define these classes formally and summarize their relations.

We start with hypergraphs having a totally unimodular incidence matrix, where a matrix is called totally unimodular if the determinant of any square submatrix is equal to 0, 1, or -1.

Definition 1.18 (Unimodular hypergraph)**.** A hypergraph is *unimodular* if its incidence matrix is totally unimodular.

The definition of a unimodular hypergraph directly implies that a hypergraph H is unimodular if and only if its dual hypergraph H^* is unimodular. Namely, the incidence matrix of H^* is equal to the transpose of the incidence matrix of H. As the determinants of square submatrices of a matrix and that of its transpose are the same, a matrix is totally unimodular if and only if its transpose is totally unimodular.

There exists a more combinatorial characterization of totally unimodular matrices than the one via determinants, see [Ghouila-Houri, 1962]. Rewritten in terms of hypergraph we obtain the following condition for a hypergraph to be unimodular.

Theorem 1.19 ([Ghouila-Houri, 1962])**.** *A hypergraph H is unimodular if for every $S \subseteq V(H)$ there exists a partition $S = S_1 \cup S_2$ such that $|e \cap S_1| - |e \cap S_2| \in \{0, \pm 1\}$ for all $e \in E(H)$. Such a partition is called an* equitable coloring *of the subhypergraph $H(S)$ restricted to S.*

A class of "bipartite hypergraphs" motivated by property (b) are so-called balanced hypergraphs defined by Berge in [Berge, 1970].

Definition 1.20 (Balanced hypergraph)**.** A hypergraph is *balanced* if it contains no strong odd cycle.

Theorem 1.19 and the definition of a balanced hypergraph imply that every unimodular hypergraph is balanced. Suppose to the contrary that H is a unimodular hypergraph that is not balanced. Then H has a strong odd cycle C. If S is the set of vertices of C, then $H(S)$ has no equitable 2-coloring, which is a contradiction to Theorem 1.19.

On the other hand, every balanced hypergraph with hyperedges of size at most three is unimodular, see Theorem 7 and the corollary thereafter in Chapter 5 of

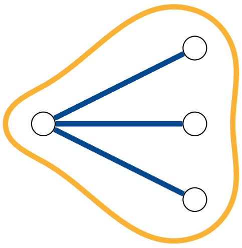


Figure 1.7: A balanced hypergraph that is not unimodular.

[Berge, 1984]. However, there are balanced hypergraphs of rank four and larger that are not unimodular. One such hypergraph is shown in Figure 1.7. It has no strong odd cycle but it also has no equitable 2-coloring.

The famous theorem of Kőnig states that the maximum size of a matching equals the minimum size of a vertex cover in a bipartite graph, see [Kőnig, 1934]. Berge and Las Vergnas proved that this min-max relation also holds for balanced hypergraphs.

Theorem 1.21. *[Berge and Las Vergnas, 1970] A hypergraph is balanced if and only if $\tau_E(H') = \nu_E(H')$ for all partial subhypergraphs H' of H.*

This theorem implies that balanced hypergraphs are closed under taking subgraphs. This follows also directly from the definition of a balanced hypergraph. Namely, a strong odd cycle in a partial subhypergraph of a balanced hypergraph H would give a strong odd cycle in H.

Corollary 1.22. *Every partial subhypergraph of a balanced hypergraph is balanced.*

Berge also proved that the dual of a balanced hypergraph is balanced, see Proposition 5 in Chapter 5 of [Berge, 1984]. This result follows from the correspondence of strong cycles in a hypergraph and induced cycles in its bipartite representation. Namely, a hypergraph H contains a strong odd cycle if and only if $\mathrm{Bip}(H)$ has an induced cycle of length $4k + 2$ for some $k \in \mathbb{N}$. As the bipartite representation of H is isomorphic to that of its dual, we directly obtain the following theorem.

Theorem 1.23. *[Berge, 1984] A hypergraph H is balanced if and only if its dual H^* is balanced.*

A stable set in a hypergraph corresponds to a matching in its dual, and a hyperedge cover corresponds to an E-vertex cover in the dual. Therefore, the previous two theorems imply the following corollary.

Corollary 1.24. *The maximum size of a stable set equals the minimum size of a hyperedge cover in a balanced hypergraph.*

Conforti, Cornuéjols, and Rao developed a decomposition theory for balanced hypergraphs leading to a polynomial time recognition algorithm for those hypergraphs, see [Conforti et al., 1999]. These results were generalized to $0, \pm 1$- matrices, see [Conforti et al., 2001a] and [Conforti et al., 2001b]. For more results on balanced matrices and the generalization to $0, \pm 1$-matrices we refer to the survey [Conforti et al., 2006] on balanced matrices.

Now, we introduce the class of normal hypergraphs. Recall that in a bipartite graph G the edge set can be partitioned into $\Delta(G)$ matchings, where $\Delta(G)$ is the maximum degree of G. This fact was proven by Kőnig in one of his first papers on graph theory, see [Kőnig, 1916].

Definition 1.25 (Normal hypergraph)**.** A hypergraph H is *normal* if and only if for every partial hypergraph H' of H the hyperedge set $E(H')$ of H' can be partitioned into $\Delta(H')$ matchings.

By definition, every partial hypergraph of a normal hypergraph is normal. In contrast to balanced hypergraphs, the class of normal hypergraphs is not closed under taking subgraphs and hypergraph duality. However, Theorem 1.21 holds if one replaces "partial subhypergraphs" with "partial hypergraphs".

Theorem 1.26. *[Lovász, 1972] A hypergraph H is normal if and only if $\tau_E(H') = \nu_E(H')$ for every partial hypergraph H' of H.*

Theorem 1.26 together with Theorem 1.21 shows that balanced hypergraphs are a subclass of normal hypergraphs. At the end of this section we show that strict containment holds.

Another generalization of bipartite graphs to hypergraphs are Mengerian hypergraphs (also called Max-Flow-Min-Cut hypergraphs), which are hypergraphs satisfying a weighted variant of Kőnig's Theorem.

Definition 1.27 (Mengerian hypergraph)**.** A hypergraph H is *Mengerian* if the system

$$\sum_{v \in e} y_v \geq 1 \ \forall e \in E(H), y \geq 0 \tag{1.1}$$

is totally dual integral (TDI).

As system (1.1) is TDI, it defines an integral polyhedron. The reverse implication is not true. For example, the dual hypergraph of K_4, which is the hypergraph with a vertex for every edge of K_4 and hyperedges corresponding to $\delta(v)$ for every vertex v of K_4, has matching number one, whereas each vertex cover has size at least two, thus, it is not Mengerian. However, system (1.1) defines an integral polyhedron.

For an interpretation in terms of hypergraphs we need the notion of vertex-expansion.

Definition 1.28 (Vertex expansion)**.** Let H be a hypergraph and $\lambda \in \mathbb{Z}_{\geq 0}$ be an integer. Expanding a vertex $v \in V(H)$ by λ means replacing v by λ new vertices $v^1, v^2, \ldots, v^\lambda$ and each hyperedge e containing v by λ new hyperedges $e^1 = e \setminus \{v\} \cup \{v^1\}$, …, $e^\lambda = e \setminus \{v\} \cup \{v^\lambda\}$. If $\lambda = 0$, then we delete v and all hyperedges e containing v. Given a function $f : V(H) \to \mathbb{Z}_{\geq 0}$, the hypergraph obtained by expanding each vertex v by $f(v)$ is denoted by H^f.

Using this notion Definition 1.27 can be restated as follows:

Observation 1.29. *A hypergraph H is Mengerian if and only if $\tau_E(H^f) = \nu_E(H^f)$ for every vertex expansion H^f.*

Observe that $\tau_E(H^f)$ is equal to the maximum f-weight of a vertex cover and $\nu_E(H^f)$ equals the maximum size of an f-matching, i.e., a multiset of hyperedges of H meeting each vertex v at most $f(v)$-times. In particular, the maximum size of a matching equals the minimum size of an E-vertex cover in Mengerian hypergraphs.

Fulkerson, Hoffman, and Oppenheim show that every balanced hypergraph H satisfies $\tau(H^f) = \nu(H^f)$ for every $f : V(H) \to \mathbb{Z}_{\geq 0}$, see [Fulkerson et al., 1974]. Thus, every balanced hypergraph is Mengerian. We construct Mengerian hypergraphs that are not balanced at the end of this section.

A special property of Mengerian hypergraphs, which we will use in Chapter 3, is that an r-uniform Mengerian hypergraph is also r-partite. For balanced hypergraphs this follows from Corollary 2 in Chapter 5 of [Berge, 1984]. For the larger class of Mengerian hypergraphs we could not find any references for this fact and thus give a short proof of it.

Theorem 1.30. *Every Mengerian r-uniform hypergraph is r-partite.*

Proof. We prove the claim via induction on r. For $r = 1$ it is trivially true.

Now, let $r \geq 2$ and suppose that the claim of the theorem holds for all $(r-1)$-uniform Mengerian hypergraphs. We define a vector $f \in Z_{\geq 0}^{V(H)}$ by $f_v := \deg_H(v)$ for all $v \in V(H)$. The vector $y \equiv \frac{1}{r}$ satisfies $\sum_{v \in e} y_v = 1$ for all $e \in E(H)$. Therefore, $\min\{f^T y : A(H)^T y \geq 1, y \geq 0\}$ is at most $\frac{1}{r} \sum_{v \in V(H)} \deg_H(v) = |E(H)|$. On the other hand, the dual of this linear program is $\max\{1^T x : A(H)x \leq f, x \geq 0\}$. The optimal value of this linear program is at least $|E(H)|$ because the vector $x \equiv 1$ satisfies $A(H)x = 1$. Therefore, $\min\{\sum_{v \in V(H)} f^T y : A(H)^T y \geq 1, y \geq 0\} = |E(H)|$ by linear programming duality. As the system $A(H)^T y \geq 1$ is totally dual integral, there exists an integral vector $y^* \in \mathbb{Z}_{\geq 0}^{V(H)}$ with $A(H)^T y^* \geq 1$ and $f^T y^* = |E(H)|$. The vector y^* only takes values $0, 1$ and thus the vertices $v \in V(H)$ with $y^*_v = 1$ form an E-vertex cover C of H. This vertex cover intersects each edge of H exactly once as otherwise $f^T y^* = \sum_{v \in C} f(v) = \sum\limits_{v \in C} \deg_H(v) = \sum_{e \in E(H)} |e \cap C| > |E(H)|$ would follow. The set C together with an $(r-1)$-partition of $H \setminus C$ forms an r-partition of H. □

Borndörfer and Heismann used a completely different approach to generalize bipartite graphs to hypergraphs in [Borndörfer and Heismann, 2012]. They look at hypergraphs with the property that the vertex set is partitioned into two subsets and each hyperedge intersects both subsets in the same number of vertices. It is possible that the two vertex subsets can be further partitioned. This leads to the definition of a so-called partitioned hypergraph.

Definition 1.31 (Partitioned Hypergraph)**.** A hypergraph H is a *partitioned hypergraph* if $V(H)$ can be partitioned into subsets U, W such that $|e \cap U| = |e \cap W|$ for all $e \in E(H)$. A partition of U and W into subsets $U_1, \dots, U_k$, $W_1, \dots, W_l$ is *feasible* if for every hyperedge $e \in E(H)$ there exist indices $i \in [k], j \in [l]$ such that $e \subseteq U_i \cup W_j$. In this case, the sets $U_1, \dots, U_k$, $W_1, \dots, W_l$ are called *parts* of H.

The trivial partition $U_1 = U$, $W_1 = W$ is always feasible, and there are a lot of feasible partitions in general. However, it is clear that every partitioned hypergraph has a unique finest feasible partition. The *part size* of a partitioned hypergraph is the maximum size of a part in its finest partition.

Partitioned hypergraphs have nothing to do with the classes of "bipartite hypergraphs" defined before. In general, they do not have totally unimodular incidence matrices, they can contain strong odd cycles, their edge set cannot be partitioned into $\Delta(H)$ matchings, and Kőnig's Theorem does not hold. It turns out that the matching problem is not tractable on partitioned hypergraphs. It is $\mathcal{NP}$-complete to decide whether a partitioned hypergraph has a perfect matching even if all parts are of size at most two, see [Borndörfer and Heismann, 2012]. We include this hypergraph class into this section as we show in Chapter 2 that there exists an approximation algorithm for finding a maximum weight matching with approximation factor depending on the part size.

We conclude this chapter with a schematic overview of the classes of "bipartite hypergraphs" we considered in this section, see Table 1.1. All of the six hypergraph classes shown in this table coincide in the 2-uniform case with the class of bipartite graphs. For arbitrary hyperedge sizes, only the implications depicted (and their transitive closure) hold. For example, the hypergraph depicted in Figure 1.7 is balanced but not unimodular. It remains to show that there are normal hypergraphs that are neither balanced nor Mengerian, and that there are Mengerian hypergraphs that are neither normal nor balanced.

Normal hypergraphs are closed under hyperedge deletion, i.e., if H is normal, then $H[E(H) \setminus F]$ is normal for all $F \subseteq E(H)$, and Mengerian hypergraphs are closed under vertex deletion, i.e., if H is Mengerian, then $H \setminus S$ is Mengerian for all $S \subseteq V(H)$. However, normal hypergraphs are not closed under vertex deletion and Mengerian hypergraphs not under hyperedge deletion. Using this fact, we construct non-normal Mengerian and non-Mengerian normal hypergraphs.

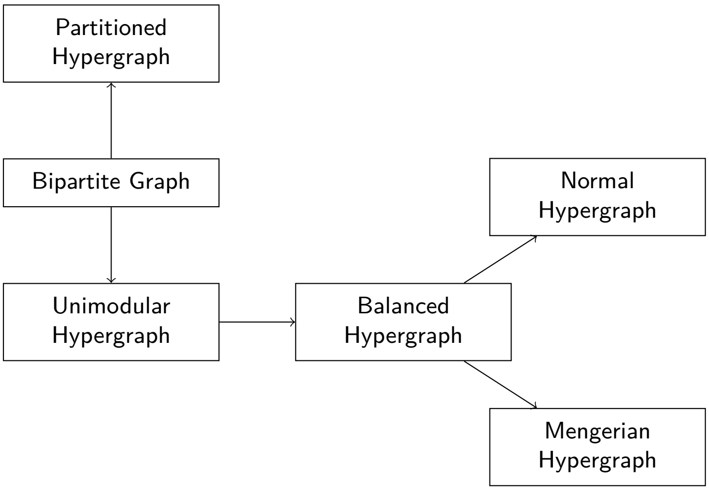


Table 1.1: Summary of the relation between hypergraphs generalizing bipartite graphs

If H is an arbitrary hypergraph and H' is the hypergraph obtained from H by adding all singleton hyperedges $\{v\}$ to the hyperedge set, then the resulting hypergraph is Mengerian. In this case system (1.1) reduces to $y_v \geq 1$ for all vertices $v \in V(H)$, which is clearly totally dual integral. If H is neither normal nor balanced, then the same is true for H' because H is a partial hypergraph of H'.

Similar, there is an easy way to construct non-balanced, non-Mengerian, normal hypergraphs. For an arbitrary hypergraph H we define an auxiliary hypergraph H' by $V(H') := V(H) \cup \{v^*\}$ and $E(H') := \{e \cup \{v^*\} : e \in E(H)\}$. Informally speaking, we add a new vertex and put this vertex into every hyperedge. The resulting hypergraph is normal because $\Delta(H'[E']) = |E'|$ for every $E' \subseteq E(H')$ and $H'[E']$ can trivially partitioned into $|E'|$ matchings. If we start with a non-balanced, non-Mengerian hypergraph H, then H' is non-balanced and non-Mengerian because the two properties are closed under vertex deletion and $H' \setminus v^* = H$.

Chapter 2

Matchings in Hypergraphs Generalizing Bipartite Graphs

Hall's and Kőnig's theorem are one of the most fundamental results in graph theory and combinatorial optimization. The former one gives a necessary and sufficient condition for the existence of a perfect matching in bipartite graphs, and the latter states that the cardinality of a maximum matching equals the cardinality of a minimum vertex cover. Kőnig's theorem can be seen as an optimization version of Hall's theorem. It turns out that both theorems can be generalized to hypergraphs, at least to some special classes.

In the first section of this chapter we review known results about matchings in hypergraphs. Afterwards, in Section 2.2, we consider partitioned hypergraphs and give a refined bound on the integrality gap of their fractional matching polytope depending on the maximum part size. We compare this approximation result to similar ones on hypergraphs with hyperedges of bounded size, and show that it yields a better approximation guarantee in some cases. A short version of Section 2.2 was published in [Beckenbach and Borndörfer, 2016].

The topic of the remaining section is the relationship of Hall's and Kőnig's theorem in graphs and hypergraphs. In the graph case, both theorems are equivalent in the sense that one can be easily proven using the other. However, we look at them from a different perspective by characterizing the graphs G that satisfy Hall's and Kőnig's theorem, respectively. In the case of Kőnig's theorem, graphs in which the maximum size of a matching equals the minimum size of a vertex cover are called Kőnig-Egerváry graphs and were characterized by Deming [Deming, 1979] and Sterboul [Sterboul, 1979]. However, graphs satisfying Hall's theorem have not received much attention. We characterize them in terms of their Gallai-Edmonds decomposition. Furthermore, we consider variants of Kőnig's and Hall's theorem for hypergraphs. In particular we give a generalization of the known Hall-type theorem for balanced hypergraphs of Conforti et al. [Conforti et al., 1996] to the larger class of normal hypergraphs. The results of Section 2.3 are published in [Beckenbach and Borndörfer, 2018].

2.1 Literature Overview

A lot of combinatorial problems can be reformulated in terms of matchings and coverings in hypergraphs, see for example [Cornuéjols, 2001]. Therefore, both concepts form a very general framework and it is clear that the theory of matchings in hypergraphs must be more involved than in the graph case. Finding a maximum weight matching in a graph is polynomial time solvable [Edmonds, 1965b]. In contrast to graphs, finding a maximum size matching in a general hypergraph is $\mathcal{NP}$-hard. Indeed, finding a perfect matching in a 3-partite hypergraph is one of Karp's 21 $\mathcal{NP}$-complete problems [Karp, 1972].

In the following we focus on four different themes covering most of the existing literature on matchings and coverings in hypergraphs, namely:

- polyhedral investigations of the matching and covering polytopes (2.1.1),
- approximation algorithms (2.1.2),
- special classes of hypergraphs (2.1.3),
- conditions for the existence of perfect matchings (2.1.4).

2.1.1 Linear Programming Methods

In this subsection we summarize how linear programming can be used to derive bounds on the matching and vertex cover numbers.

In the introductory chapter we have seen that the size of a maximum matching is at most the minimum size of a vertex cover (for both the vertex and hyperedge version). In an r-uniform hypergraph H we have $\tau_E(H) \leq r\nu_E(H)$ as the union of the vertices in an E-maximum matching forms an E-vertex cover. A famous conjecture states that $\tau_E(H) \leq (r-1)\nu_E(H)$ if H is r-partite and $r \geq 2$. This conjecture appeared in the PhD-Thesis of Henderson, see Conjecture 4.3 in [Henderson, 1971], and is often called Ryser's conjecture after Herbert John Ryser, who was the advisor of Henderson. It is known that Ryser's conjecture is true for $r \leq 3$. Namely, for $r = 2$ it is just Kőnig's theorem and for $r = 3$ it was proven by Aharoni [Aharoni, 2001]. For $r \geq 4$ Ryser's conjecture is still open. However, there are some results concerning fractional matchings and fractional vertex covers.

To define fractional matchings and fractional vertex covers, we first observe that the maximum b-weight matching and the b-vertex cover problem on a hypergraph H can be formulated as integer linear programs as follows:

$$
\begin{array}{llll@{\qquad}llll}
\max & \displaystyle\sum_{e\in E(H)} b(e)x_e & & & \min & \displaystyle\sum_{v\in V(H)} y_v & & \\
\text{s.t.} & \displaystyle\sum_{e\in\delta_H(v)} x_e \leq 1 & & \forall v\in V(H) & \text{s.t.} & \displaystyle\sum_{v\in e} y_v & \geq b(e) & \forall e\in E(H) \\
 & x_e \in \{0,1\} & & \forall e\in E(H) & & y_v & \in\{0,1\} & \forall v\in V(H).
\end{array}
$$

Replacing the binary constraints by $x_e \geq 0$ and $y_v \geq 0$ we obtain two linear programs whose solutions are called *fractional matchings* and *fractional b-vertex covers*, respectively. Furthermore, we call the polytope given by the inequalities of type $x(\delta(v)) \leq 1$ and $x \geq 0$ the *fractional matching polytope* and denote it by $\mathcal{FP}(H)$. The maximum b-weight of a fractional matching is denoted by $\nu_b^*(H)$ and the minimum size of a fractional b-vertex cover by $\tau_b^*(H)$. For the special case that $b(e) = 1$ we denote $\nu_b^*(H)$ by $\nu_E^*(H)$ and $\tau_b^*(H)$ by $\tau_E^*(H)$.

By duality, $\nu_b^*(H) = \tau_b^*(H)$ holds, and this value gives an upper bound on $\nu_b(H)$ and a lower bound on $\tau_b(H)$. In general, the gap between $\nu_b^*(H)$ and $\nu_b(H)$, or $\tau_b^*(H)$ and $\tau_b(H)$ can be arbitrarily large. However, Füredi [Füredi, 1981] showed that the ratio $\nu_E^*(H)/\nu_E(H)$ is bounded by $(r - 1 + {}^1\!/r)$ if the maximum size of a hyperedge is r. Furthermore, the constant $(r - 1 + {}^1\!/r)$ can be improved to $(r - 1)$ if H does not contain a partial hypergraph that is a finite projective plane, where a finite projective plane is an r-uniform, r-regular hypergraph on $r^2 - r + 1$ vertices, and $r^2 - r + 1$ hyperedges for some $r \in \mathbb{N}$, $r \geq 2$, satisfying the following conditions (see for example Sect. 2 in Ch. 2 of [Berge, 1984]):

(1) for two distinct vertices v, w there exists exactly one hyperedge containing both vertices,

(2) two distinct hyperedges intersect in exactly one vertex.

By the last condition, every finite projective plane has matching number one. Furthermore, its fractional vertex cover number is $(r - 1 + {}^1\!/r)$. Thus, if H is a finite projective plane of rank r, than $\nu_E^*(H) = \tau_E^*(H) = (r - 1 + {}^1\!/r)\nu_E(H)$.

Later, Füredi, Kahn, and Seymour considered the ratio of $\nu_b^*(H)$ and $\nu_b(H)$ for general weight functions b.

Theorem 2.1. *[Füredi et al., 1993] If H is a hypergraph of rank r, where $r \geq 3$, and $b : E(H) \to \mathbb{Q}_{\geq 0}$ is a function, then*

$$\nu_b^*(H) \leq (r - 1 + {}^1\!/r)\nu_b(H).$$

If H does not contain a partial hypergraph that is isomorphic to a finite projective plane, then this inequality can be improved to

$$\nu_b^*(H) \leq (r - 1)\nu_b(H).$$

This result shows that a fractional variant of Ryser's conjecture holds, namely $\tau_E^*(H) \leq (r-1)\nu_E(H)$ for every r-partite hypergraph H as an r-partite hypergraph cannot contain a finite projective plane as a partial subhypergraph. Namely, a hypergraph of rank r with at most one hyperedge is not a finite projective plane. Furthermore, any partial hypergraph of an r-partite hypergraph is r-partite, and two vertices of the same vertex class are not contained in a common hyperedge, thus (1) is not satisfied by any partial hypergraph of an r-partite hypergraph with more than one hyperedge.

Chan and Lau give a constructive proof of Theorem 2.1, which directly gives a polynomial time approximation algorithm for the weighted matching problem in uniform hypergraph, see [Chan and Lau, 2012]. Furthermore, for the unweighted case they show that after adding all "clique inequalities", which are inequalities of the form $\sum_{e \in Q} x_e \leq 1$ for a set Q of pairwise intersecting hyperedges, the integrality gap decreases to $\frac{r+1}{2}$.

The work of Chan and Lau shows that it is worthwhile to consider additional inequalities that are valid for (integral) matchings and cut off some fractional matchings. This is the classical cutting plane-approach. In this context the matching problem in hypergraphs is mostly called *set packing problem.* We do not go into detail and refer the interested reader to [Borndörfer, 1998] or [Marchand et al., 2002].

2.1.2 Approximation Algorithms

For general hypergraphs there exists no constant factor approximation algorithm for the maximum weight matching problem unless $\mathcal{P} = \mathcal{NP}$. However, there are approximation algorithms with approximation factor depending on the number of vertices or the maximum size of a hyperedge.

In the unweighted case, Halldórsson, Kratochvíl, and Telle show that an easy greedy algorithm achieves an approximation guarantee of $\sqrt{|V(H)|}$ and that this result is best possible in the sense that there is no $\mathcal{O}(|V(H)|^{\frac{1}{2}-\epsilon})$-approximation algorithm for any $\epsilon > 0$ unless $\mathcal{P} = \mathcal{NP}$, see [Halldórsson et al., 2000]. For the weighted matching problem there exists a $2\sqrt{|V(H)|}$-approximation algorithm, see [Halldórsson, 1999].

Considering the maximum size $r(H)$ of a hyperedge in a hypergraph H as a parameter, the result of [Chan and Lau, 2012] mentioned above directly gives an $(r(H) - 1 + {}^1\!/\!_{r(H)})$-approximation algorithm based on LP-methods. On the other hand, in [Hazan et al., 2006] it is shown that no $\Omega\left({}^{r(H)}\!/\!_{\ln(r(H))}\right)$-approximation algorithm exists unless $\mathcal{P} = \mathcal{N}P$. The best known approximation algorithms for the unweighted case have a performance guarantee of ${}^{(r(H)+1)}\!/\!_3 + \epsilon$ and are based on local search, see [Cygan, 2013] and [Fürer and Yu, 2014]. For the weighted matching problem an $({}^{(r(H)+1)}\!/\!_2 + \epsilon)$-approximation algorithm exists [Berman, 2000]. This result is stated for the maximum weight independent set problem in d-claw free

parameter	unweighted	weighted	hardness
$\lvert V(H)\rvert$	$\lvert V(H)\rvert^{0.5}$	$2\lvert V(H)\rvert^{0.5}$	$\Omega(\lvert V(H)\rvert^{0.5-\epsilon})$
$r(H) = \max_{e\in E(H)} \lvert e\rvert$	$\frac{r(H)+1}{3} + \epsilon$	$\frac{r(H)+1}{2} + \epsilon$	$\Omega(r(H)/\ln(r(H))$

Table 2.1: Approximation guarantees.

graphs. There is a tight connection between the matching problem in hypergraphs and the independent set problem in graphs. Namely, a maximum weight matching in a hypergraph H corresponds to a maximum weight independent set in the line graph $L(H)$. The condition that the hyperedges have size at most r translates into the requirement that $L(H)$ has no $(r+1)$-claw, which is an induced subgraph consisting of a stable set of size $(r+1)$ and a "center" vertex connected to all these $(r+1)$ vertices. Therefore, approximation results for independent sets in $(r+1)$-claw free graphs translate to ones for matchings in hypergraphs with hyperedges of size at most r.

Table 2.1 gives an overview of the known approximation guarantees depending on the number of vertices or the maximum size of a hyperedge.

2.1.3 Hypergraphs Generalizing Bipartite Graphs

In this subsection we investigate the hypergraph classes generalizing bipartite graphs that were introduced in Chapter 1. These hypergraphs have nice polyhedral properties, which has consequences for the matching and vertex cover problem restricted to them.

We start with unimodular hypergraphs, which are by definition hypergraphs with a totally unimodular incidence matrix. On totally unimodular matrices a lot of integer optimization problems can be solved by linear programming as the famous Hoffman-Kruskal theorem shows.

Theorem 2.2. *[Hoffman and Kruskal, 2010] A matrix $A \in \mathbb{Q}^{n\times m}$ is totally unimodular if and only if for all integral vectors $b \in \mathbb{Z}^n$ and $c \in \mathbb{Z}^m$ the polyhedron*

$$Q(b,c) := \{x \in \mathbb{Q}^m : Ax \geq b, x \geq c\}$$

has only integral vertices.

There are a lot of characterizations for totally unimodular matrices. In particular the Hoffman-Kruskal theorem is also equivalent to:

For all $b', c' \in \mathbb{Z}^n$, $p, q \in \mathbb{Z}^m$ the polytope

$$Q(b',c',p,q) := \{x \in \mathbb{Q}^m : b' \leq Ax \leq c', p \leq x \leq q\}$$

is empty or has only integral vertices.

This implies that finding a maximum weight matching or minimum weight vertex cover in a unimodular hypergraph is polynomial-time solvable by linear programming. However, a combinatorial algorithm running in polynomial time is only known for a subclass of unimodular hypergraphs, so-called restricted unimodular hypergraphs.

Definition 2.3 (Restricted unimodular hypergraph)**.** A hypergraph is *restricted unimodular* if and only if it has no odd cycle.

Restricted unimodular hypergraphs are defined in terms of their incidence matrices in [Yannakakis, 1985], where also a polynomial time recognition and matching algorithm is given. Both algorithms are based on a decomposition of the incidence matrix of a restricted unimodular hypergraph into incidence matrices of bipartite graphs and directed graphs. A combinatorial algorithm for the maximum cardinality matching problem in restricted unimodular hypergraphs is given in [Conforti and Cornuéjols, 1987]. Furthermore, Crama, Hammer, and Ibaraki give a decomposition based algorithm for matching and covering problems in so-called strongly unimodular hypergraphs, which is a subclass of unimodular hypergraphs containing the class of restricted unimodular hypergraphs, see [Crama et al., 1990]. In [Crama et al., 1990] it is also mentioned that there is an unpublished algorithm of Edmonds and Bland for matching and covering problems in unimodular hypergraphs based on Seymour's decomposition of totally unimodular matrices [Seymour, 1980]. However, this result seems to remain unpublished. In the meantime, Artmann, Weißmantel, and Zenklusen developed a polynomial time algorithm for optimizing a linear function over matrices in which all subdeterminants are bounded by two in absolute value, see [Artmann et al., 2017]. Their algorithm is based on Seymour's decomposition of totally unimodular matrices, and gives also a decomposition based algorithm for matching and covering problems on unimodular hypergraphs.

For balanced hypergraphs the polyhedron $Q(b,c,p,q)$ might have fractional vertices. However, for some values of b,c,p,q it is integral.

Theorem 2.4. *[Fulkerson et al., 1974] If H is balanced, then the following polytopes are empty or have only integral vertices:*

- $\{x \in \mathbb{Q}^{E(H)} : x(\delta(v)) \leq 1 \ \forall v \in V(H), x \geq 0\}$,
- $\{x \in \mathbb{Q}^{E(H)} : x(\delta(v)) = 1 \ \forall v \in V(H), x \geq 0\}$,
- $\{x \in \mathbb{Q}^{E(H)} : x(\delta(v)) \geq 1 \ \forall v \in V(H), x \geq 0\}$.

It turns out that normal hypergraphs can be characterized as those hypergraphs that have an integral fractional matching polytope.

Theorem 2.5. *[Lovász, 1972] A hypergraph H is normal if one of the following equivalent condition holds:*

(a) the fractional matching polytope $\{x \in \mathbb{Q}^{E(H)} : x(\delta(v)) \leq 1 \ \forall v \in V(H), x \geq 0\}$ has only integral extreme points,

(b) the system $x(\delta(v)) \leq 1$ for all $v \in V(H)$, $x \geq 0$ is total dual integral.

The equivalence of (a) and (b) is notable as in general it is not true that a set of inequalities defining an integral polyhedron also defines a TDI-system, though the reverse direction does hold. Statement (b) also implies that a normal hypergraph satisfies $\nu_b(H) = \nu_b^*(H) = \tau_b^*(H) = \tau_b(H)$ for all $b \in \mathbb{Q}_{\geq 0}^{E(H)}$, i.e., normal hypergraphs satisfy a weighted variant of Kőnig's theorem.

By Theorem 2.4 and Theorem 2.5 maximum weight matchings in balanced or normal hypergraphs can be calculated in polynomial time using LP-methods. However, there are no combinatorial polynomial time matching algorithms known for normal or balanced hypergraphs.

Table 2.2 gives an overview about the type of polynomial time algorithms known for finding a maximum size matching in the different classes of hypergraphs generalizing bipartite graphs.

2.1.4 Hall- and Dirac-Type Theorems

Hall's theorem gives a good characterization for the non-existence of a perfect matching in the sense that if a bipartite graph G has no perfect matching, then we can certify this by giving a stable set S with less than $|S|$ neighbors. There are several attempts to generalize this result to hypergraphs starting with a conjecture of [Aharoni and Kessler, 1990], which turned out to be true. Its proof can be found in [Aharoni and Haxell, 2000]. They look at hypergraphs H with the property that the vertex set can be partitioned into two sets A and B such that $|e \cap A| = 1$ for all hyperedges $e \in E(H)$. They call such a hypergraph a *bipartite hypergraph* (though this hypergraphs have nothing to do with the ones we considered in the previous section). For a bipartite hypergraph H on $A \cup B$ and a subset $C \subseteq A$ the hypergraph H_C has vertex set B and hyperedges $E(H_C) = \{e \cap B : e \in E(H), e \cap B \neq \emptyset, e \cap C \neq \emptyset\}$.

Theorem 2.6 ([Aharoni and Haxell, 2000])**.** *Let H be a bipartite hypergraph on $A \cup B$ with hyperedges of size $r \in \mathbb{N}$. If $\nu_E(H_C) > (r-1)(|C|-1)$ for all $C \subseteq A$, then $\nu_E(H) = |A|$.*

Observe that a matching of H contains at most $|A|$ hyperedges as every hyperedge intersects A in exactly one vertex, i.e., Theorem 2.6 gives a sufficient condition for the existence of a matching of maximum possible size. In the graph case, $r = 2$,

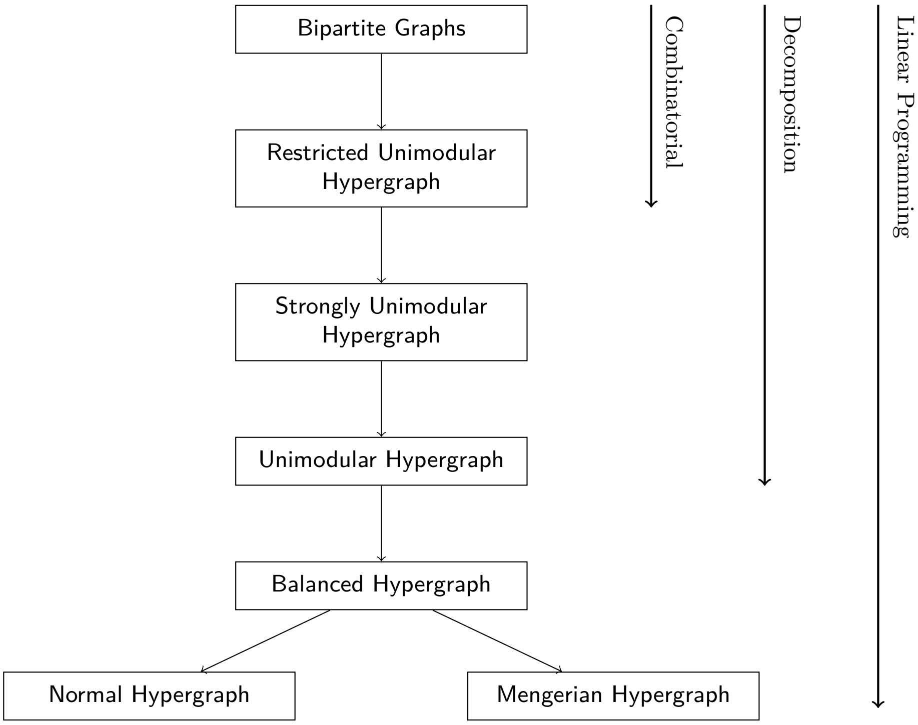


Table 2.2: Types of the known polynomial time matching algorithms for hypergraph classes generalizing bipartite graphs.

it reduces to Hall's condition $|N(C)| \geq |C|$ as $\nu(H_C)$ is equal to the number of neighbors of C in B.

A similar generalization of Hall's theorem is given by Haxell.

Theorem 2.7. *[Haxell, 1995] Let H be a bipartite hypergraph on $A \cup B$ with hyperedges of size r. If $\tau_E(H_C) > (2r-3)(|C|-1)$ for all $C \subseteq A$, then H has a matching of size $|A|$.*

The proof of this theorem in [Haxell, 1995] is non-constructive and does not give rise to an efficient algorithm to find a matching of size $|A|$. In [Annamalai, 2016] it is shown that a polynomial time algorithm can be designed if the slightly stronger condition $\tau_E(H_C) > (2r-3+\epsilon)(|C|-1)$ for some fixed $\epsilon > 0$ holds.

Finally, in 2000 Aharoni and Haxell gave a Hall-type theorem for matchings in hypergraphs implying Theorem 2.6, see [Aharoni and Haxell, 2000]. They characterize when for a family $\mathcal{A}$ of hypergraphs a function $f : \mathcal{A} \to \bigcup_{H \in \mathcal{A}} E(H)$ exists with the properties that $f(H) \in E(H)$ for all $H \in \mathcal{A}$ and $f(H) \cap f(H') = \emptyset$ for any pair of distinct hypergraphs $H, H' \in \mathcal{A}$. They call a function f with the properties above a *system of disjoint representatives* of $\mathcal{A}$. Observe that if all hypergraphs are 1-uniform, then a system of disjoint representatives of $\mathcal{A}$ is the same as a system of distinct representatives of the family $\{V(H) : H \in \mathcal{A}\}$. Hall's theorem was originally stated as a condition for a set system to have a system of distinct representatives. Thus, the result of Aharoni and Haxell generalizes Hall's theorem in its original form.

Another direction for generalizing Hall's theorem to hypergraphs was pursued by Conforti, Cornuéjols, Kapoor and Vušković, who focused on balanced hypergraphs in [Conforti et al., 1996]. As a motivation, we reformulate Hall's theorem for bipartite graphs. It states that a bipartite graph G has no perfect matching if and only if there exists a set S of vertices with less than $|S|$ neighbors. If we color the vertices in S blue and all neighbors of S red, then each edge contains at least as many red vertices as blue ones. Thus, Hall's theorem can be reformulated as follows:

> A bipartite graph has no perfect matching if and only if there exist disjoint vertex sets B and R such that $|B| > |R|$ and each edge contains at least as many vertices in R as in B.

This reformulation can be used to give a Hall-type theorem for balanced hypergraphs.

Theorem 2.8. *[Conforti et al., 1996] A balanced hypergraph has no perfect matching if and only if there exists disjoint vertex sets B and R such that $|B| > |R|$ and each hyperedge contains at least as many vertices in R as in B.*

Conforti et al. use linear programming methods to prove this theorem, whereas purely combinatorial proofs can be found in the articles [Huck and Triesch, 2002] and [Scheidweiler and Triesch, 2016]. We will show in Section 2.3.2 that Theorem 2.8 also holds for uniform hypergraphs H with the additional property that the graph G on $V(H)$ with edges $\{v, w\}$ for every $v, w \in V(H)$ such that there exists a hyperedge $e \in E(H)$ with $v, w \in e$ is perfect. Furthermore, there is a variant for normal hypergraphs for which we allow to take multiple copies of some vertices into the sets R and B in order for a similar statement as that in Theorem 2.8 to hold.

In the remainder of this section we focus on the connection between the degree of the vertices of a hypergraph and the existence of a perfect matching. Dirac's theorem states that every graph on n vertices of minimum degree at least $n/2$ has a Hamiltonian cycle. In particular, a perfect matching exists in the case that n is even. The complete bipartite graph on n vertices with one vertex class of size $n/2 + 1$ and the other of size $n/2 - 1$ shows that the bound $n/2$ on the minimum degree is best possible.

In the hypergraph case, the literature focuses on r-uniform or r-partite hypergraphs. Furthermore, the minimum degree is defined slightly different. In an r-uniform hypergraph H the degree of a set of l vertices $\{v_1, v_2, \ldots, v_l\}$ is the number of hyperedges of H containing $\{v_1, v_2, \ldots, v_l\}$, and the minimum l-degree, $\delta_l(H)$, is the minimum degree of a set of l vertices. Of particular interest is the minimum $(r-1)$-degree: $\delta_{r-1}(H)$. Rödl, Ruciński, and Szemerédi give an exact bound on $\delta_{r-1}(H)$ in order for H to contain a perfect matching, see [Rödl et al., 2009]. Keevash and Mycroft investigate r-uniform hypergraphs with an $(r-1)$-degree of at least $\frac{n}{r}$ and characterize the obstructions for the existence of a perfect matching [Keevash and Mycroft, 2015]. Using this theory they develop a polynomial time algorithm for the perfect matching problem on r-uniform hypergraphs of minimum $(r-1)$-degree at least $(1/r + \epsilon)|V(H)|$ for each fixed $\epsilon > 0$ in [Keevash et al., 2015].

The results of [Rödl et al., 2009] and [Keevash and Mycroft, 2015] use probabilistic arguments. In contrast, in [Aharoni et al., 2009] a $(r-1)$-degree condition for the existence of perfect matching in r-partite graphs is given that is derived by elementary combinatorial arguments. In an r-partite hypergraph H with r-partition $V_1, \ldots, V_r$ the $(r-1)$-degree is defined as the minimum degree of all sets S of size $r-1$ such that S contains at most one vertex from each vertex class V_i.

There are a lot more Dirac-Type results for hypergraphs, e.g., bounds on the l-degree for $l < r-1$ forcing a hypergraph to have a perfect matching or results on the existence of Hamiltonian cycles. We refer the interested reader to the survey article [Zhao, 2016].

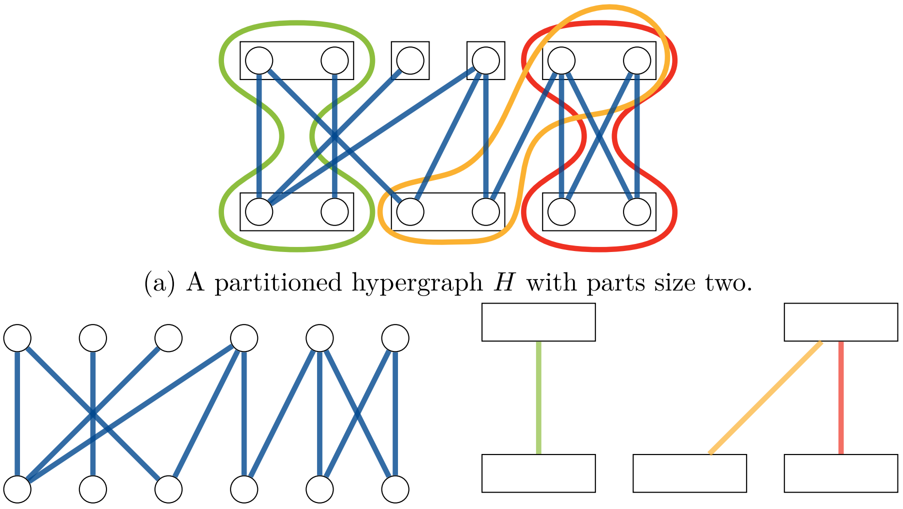
2.2 Matchings in Partitioned Hypergraphs

This section deals with matchings in partitioned hypergraphs. We give a refined bound of the ratio between $\nu_b^*(H)$ and $\nu_b(H)$ in those hypergraphs depending on the maximum part size. The proof methods are constructive and yield polynomial time approximation algorithms for the maximum weight matching problem.

The fact that the perfect matching problem is $\mathcal{NP}$-hard on partitioned hypergraphs implies that there exists no polynomial time constant factor approximation algorithm for finding a maximum weight matching unless $\mathcal{P} = \mathcal{NP}$ as such an algorithm can be used to design a polynomial time algorithm to decide whether a partitioned hypergraph has a perfect matching. However, we show that the matching problem on partitioned hypergraphs admits an approximation algorithm whose approximation factor depends on the part size.

A partitioned hypergraph H with parts of size one is just a bipartite graph. In this case the matching problem is polynomial time solvable and $\nu_b^*(H) = \nu_b(H)$ holds for all $b \in \mathbb{Z}_{\geq 0}^{E(H)}$.

Next, we consider the case that all parts have size at most two, and there is at least one part of size equal to two. In this case, all hyperedges have size two or four, and the latter connect two parts of size two. This implies that a partitioned hypergraph with parts of size at most two can be seen as a superposition of two



(a) A partitioned hypergraph H with parts size two.

(b) The bipartite graph H_1 induced by the edges of H.

(c) The bipartite graph H_2 on the parts induced by the hyperedges of size four.

Figure 2.1: Decomposition of a partitioned hypergraph with parts of size two into a pair of bipartite graphs.

bipartite graphs. One graph is obtained by restricting the hypergraph to the set of all edges (= hyperedges of size two). The second graph has a vertex for each part of size two and an edge between two vertices if a hyperedge of size four connects the corresponding parts.

Example 2.9. A partitioned hypergraph with parts of size one and two is drawn in Figure 2.1a. Vertices are drawn as cycles, its parts are indicated by rectangles around the vertices, edges are drawn as straight lines, and hyperedges as closed curves. Figure 2.1b shows the bipartite graph induced by the edges and Figure 2.1c the bipartite graph on the parts induced by the hyperedges of size four.

Using the decomposition of a partitioned hypergraph with part size two into a pair of bipartite graphs we prove that the fractional perfect matching polytope has an integrality gap of at most two.

Theorem 2.10. *Let H be a partitioned hypergraph with parts of size at most two. For every function $b : E(H) \to \mathbb{Q}$ we have $\nu_b^*(H) \leq 2\nu_b(H)$, i.e., the integrality gap of the fractional matching polytope is at most two.*

Proof. Let H_1 be the bipartite graphs induced by the hyperedges of size two of H and let the weight of the edges of H_1 be the same as in H. Furthermore, let H_2

be the bipartite graph with a vertex for each part of size two and an edge between two vertices if there exists a hyperedge of size four connecting the corresponding parts of H. The weight of an edge in H_2 is set to the weight of the corresponding hyperedge of size four in H. Every fractional matching $x \in \mathbb{Q}^{E(H)}$ decomposes into a fractional matching $x_1 \in \mathbb{Q}^{E(H_1)}$ of H_1 and a fractional matching $x_2 \in \mathbb{Q}^{E(H_2)}$ of H_2. If we choose the vector of x_1, x_2 for which $\sum_{e \in E(H)} b(e)x_i(e)$ is larger, we obtain that $2\sum_{e \in E(H_i)} b(e)x_i(e) \geq \sum_{e \in E(H)} b(e)x(e)$ for $i = 1$ or $i = 2$. As H_i is a bipartite graph there exists a matching M with $\sum_{e \in M} b(e) \geq \sum_{e \in E(H_i)} b(e)x_i(e)$. The set M corresponds also to a matching M' in H (consisting either just of edges or hyperedges of size four), and $2\sum_{e \in M'} b(e) \geq 2\sum_{e \in E(H_i)} b(e)x_i(e) \geq \sum_{e \in E(H)} b(e)x(e)$. □

The proof of the previous theorem gives a simple 2-approximation algorithm for the maximum weight matching problem in partitioned hypergraphs with maximum part size two:

> Calculate a maximum weight matching of H_1 and one of H_2 (e.g. using the Hungarian method), and output the one with the larger weight.

The upper bound of two on $\nu_b^*(H)/\nu_b(H)$ cannot be improved using our method as the following example shows.

Example 2.11. Consider the partitioned hypergraph H, whose vertex set is the disjoint union of $U := \{u_1, u_2, u_3, u_4\}$ and $W := \{w_1, w_2, w_3, w_4\}$. The hyperedges of H are $\{u_1, u_2, w_2, w_3\}$, $\{u_3, u_4, w_2, w_3\}$, $\{u_1, w_1\}$, $\{u_2, w_1\}$, $\{u_3, w_4\}$, $\{u_4, w_4\}$. The minimal partition of H is $U_1 = \{u_1, u_2\}$, $U_2 = \{u_3, u_4\}$, $W_1 = \{w_1\}$, $W_2 = \{w_2, w_3\}$, $W_3 = \{w_4\}$. This hypergraph is drawn in Figure 2.2, where hyperedges of size four are drawn as straight lines connecting the two parts they contain.

If all hyperedges of size four receive weight 2 and all edges weight 1, then $x \equiv \frac{1}{2}$ is a maximum weight fractional matching and its weight is 4. The maximum weight of a matching consisting solely of edges is 2, which is the same as the maximum weight of a matching using just hyperedges of size four. However, H has a matching of weight 3 consisting of the hyperedge $\{u_1, u_2, w_2, w_3\}$ and the edge $\{u_3, w_4\}$.

For the general case of partitioned hypergraphs with parts of size at most d we analyze the proof of Theorem 2.1 in [Chan and Lau, 2012] to derive an approximation algorithm whose approximation factor depends on d. Chan and Lau consider the set $N[e] := \{e' : e \cap e' \neq \emptyset\}$ of all hyperedges intersecting a fixed hyperedge e. The crucial point of their proof for an integrality gap of $r - 1$ for the fractional perfect matching polytope of an r-partite hypergraph is that for every extreme point x of the fractional matching polytope with $x > 0$ there exists a hyperedge $e \in E(H)$ with $x(N[e]) \leq r-1$. The further analysis of the algorithm in [Chan and Lau, 2012] does not use that the considered hypergraph is r-partite. If we can show that for every extreme point x of the fractional perfect matching polytope with $x > 0$ there exists

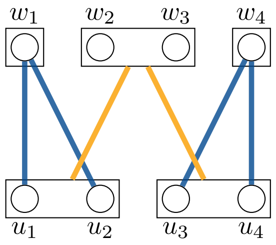


Figure 2.2: The hypergraph of Example 2.11.

a hyperedge e with $x(N[e]) \leq \alpha$ for some class $\mathcal{C}$ of hypergraphs, then the method of Chan and Lau directly gives an α-approximation algorithm for the weighted matching problem restricted to $\mathcal{C}$. For partitioned hypergraphs we prove the following bound.

Lemma 2.12. *Let H be a partitioned hypergraph with maximum part size d and let x be a fractional matching with $x_e > 0$ for all $e \in E(H)$. There exists a hyperedge $e^* \in E(H)$ with $x(N[e^*]) \leq 2\sqrt{d}$.*

Proof. If there exists a hyperedge e^* of size less than $2\sqrt{d}$, then

$$\sum_{e \in N[e^*]} x(e) \leq \sum_{v \in e^*} \sum_{e \in \delta_H(v)} x(e) \leq |e^*| < 2\sqrt{d}.$$

Otherwise, all hyperedges of H have size at least $2\sqrt{d}$. We choose $e^* \in E(H)$ arbitrarily and denote by U_i and W_j the two parts of H such that $e^* \subseteq U_i \cup W_j$. Summing over all inequalities $x(\delta(v)) \leq 1$ for $v \in U_i$ and using $|e| \geq 2\sqrt{d}$ for all $e \in E(H)$ gives

$$\sum_{e \in \delta(U_i)} \sqrt{d} x(e) \leq \sum_{e \in \delta(U_i)} \frac{|e|}{2} x(e) = \sum_{v \in U_i} x(\delta_H(v)) \leq |U_i| \leq d,$$

and the same inequality holds for W_j. Every hyperedge in $N[e^*]$ intersects e in U_i or W_j, thus $N[e^*] \subseteq \delta_H(U_i) \cup \delta_H(W_j)$. It follows that

$$\sum_{e \in N[e^*]} x(e) \leq \sum_{e \in \delta(U_i)} x(e) + \sum_{e \in \delta(W_j)} x(e) \leq 2\sqrt{d}.$$ □

Using Lemma 2.12 and the ideas of Chan and Lau we directly obtain the following approximation result for the matching problem on partitioned hypergraphs.

Theorem 2.13. *[Beckenbach and Borndörfer, 2016] If H is a partitioned hypergraph with parts of size at most d, then $\nu_b^*(H) \leq 2\sqrt{d} \cdot \nu_b(H)$.*

Proof. Observe that we can delete hyperedges of weight zero without changing the maximum weight of a fractional matching or of a matching. Thus, we can assume that $b > 0$.

We use induction on the number of hyperedges of H. If $|E(H)| \leq 1$ the claim is trivial. For the induction step let x be a fractional matching of maximum b-value. If there exists a hyperedge e^* with $x(e^*) = 0$, then look at the hypergraph H' obtained from H by deleting e^*. If we restrict x to $E(H') = E(H) \setminus \{e^*\}$, then we get a fractional matching of H'. By the induction hypothesis there exists a matching M of H' with $2\sqrt{d} \cdot b(M) \geq \sum_{e \in E(H')} b(e)x_e$. The set M is also a matching in H with $2\sqrt{d} \cdot b(M) \geq \sum_{e \in E(H)} b(e)x_e$.

If $x > 0$, then there exists a hyperedge e^* with $x(N[e^*]) \leq 2\sqrt{d}$ by Lemma 2.12. We define new weights b_1, b_2 by $b_1(e) := b(e^*)$ for all $e \in N[e^*]$ and $b_1(e) := 0$ for all $e \in E(H) \setminus N[e^*]$, and by $b_2(e) := b(e) - b_1(e)$ for all $e \in E(H)$. Hyperedges with $b_2(e) \leq 0$ can be deleted without changing the value of $\nu^*_{b_2}(H)$ or $\nu_{b_2}(H)$. We denote the resulting hypergraph by H'. As $b_2(e^*) = 0$, H' has less hyperedges than H, and x restricted to $E(H')$ is a fractional matching of H'. By the induction hypothesis there exists a matching M' of H' with $2\sqrt{d} \cdot b_2(M') \geq \sum_{e \in E(H')} b_2(e)x_e$. If $M' \cup \{e^*\}$ is a matching of H we set $M := M' \cup \{e^*\}$, otherwise we set $M := M'$. In both cases, we have $b_2(M) = b_2(M')$ and $b_1(M) \geq b(e^*)$ because $b_2(e^*) = 0$ and $N[e^*] \cap M \neq \emptyset$. It follows that

$$\begin{aligned}
2\sqrt{d} \cdot b(M) &= 2\sqrt{d} \cdot b_2(M) + 2\sqrt{d} \cdot b_1(M) \geq 2\sqrt{d} \cdot b_2(M') + 2\sqrt{d} \cdot b(e^*) \\
&\geq \sum_{e \in E(H')} b_2(e)x_e + b(e^*) \sum_{e \in N[e^*]} x(e) \\
&\geq \sum_{e \in E(H)} b_2(e)x_e + \sum_{e \in E(H)} b_1(e)x_e \\
&= \sum_{e \in E(H)} b(e)x_e.
\end{aligned}$$

Thus, we have shown that $2\sqrt{d}\nu_b(H) \geq \nu^*_b(H)$. □

The proof of Theorem 2.13 gives a polynomial time $2\sqrt{d}$-approximation algorithm for finding a maximum weight matching that is literally the same as the iterative k-dimensional matching algorithm of [Chan and Lau, 2012]. We omit details and remark that the only difference to the algorithm of Chan and Lau is that we choose a hyperedge with $x(N[e]) \leq 2\sqrt{d}$ instead of $x(N[e]) \leq k - 1$.

Corollary 2.14. *There exists a polynomial time $2\sqrt{d}$-approximation algorithm for the maximum weight matching problem on partitioned hypergraphs of part size d.*

The constant $2\sqrt{d}$ in Theorem 2.13 is almost tight as the following example shows.

0	1	2	3	4
1	2	3	4	0
2	3	4	0	1
3	4	0	1	2
4	0	1	2	3

0	2	4	1	3
1	3	0	2	4
2	4	1	3	0
3	0	2	4	1
4	1	3	0	2

0	3	1	4	2
1	4	2	0	3
2	0	3	1	4
3	1	4	2	0
4	2	0	3	1

0	4	3	2	1
1	0	4	3	2
2	1	0	4	3
3	2	1	0	4
4	3	2	1	0

Figure 2.3: Four pairwise orthogonal Latin squares of order five.

Example 2.15. Consider the partitioned hypergraph H on $U := \{u_1, u_2, u_3, u_4\}$ and $W := \{w_1, w_2, w_3, w_4\}$ with the four hyperedges $\{u_1, u_2, w_1, w_2\}$, $\{u_3, u_4, w_1, w_2\}$, $\{u_1, u_3, w_3, w_4\}$, and $\{u_2, u_4, w_3, w_4\}$. The minimal partition of H is $U_1 = U$ and $W_1 = \{w_1, w_2\}$, $W_2 = \{w_3, w_4\}$, i.e., the maximum part size is four. Each pair of hyperedges has a nonempty intersection, so the maximum size of a matching is one. Setting $x \equiv \frac{1}{2}$ gives a fractional matching because every vertex is incident to two hyperedges. The size of x is $4 \cdot \frac{1}{2} = 2 = \sqrt{4}$.

The previous example shows that we can only hope to improve the constant $2\sqrt{d}$ to $\sqrt{d}$. In the remainder of this section we generalize this example to maximum part sizes of $d = q^2$ where q is a prime power. Namely, we construct a partitioned hypergraph of part size d in which every vertex has degree q and every pair of hyperedges intersect. Then $x \equiv \frac{1}{q}$ is a fractional matching of size $q^2 \cdot \frac{1}{q} = q = \sqrt{d}$, whereas the maximum size of a matching is one. We use Latin squares for our construction.

Definition 2.16 (Latin square)**.** A *Latin square* of order n is an $n \times n$-array filled with numbers $0, 1, \ldots, n-1$ such that each number occurs exactly once in each row and each column. If we denote a Latin square by L, then $L[i, j]$ denotes the entry at row i and column j. Two Latin squares L, L' of order n are *orthogonal* if $(L[i, j], L'[i, j]) \neq (L[k, l], L'[k, l])$ for $(i, j) \neq (k, l)$, $i, j, k, l \in \{1, \ldots, n\}$.

A set of pairwise orthogonal Latin squares of order n can have size at most $n - 1$. In general, it is not known for which number n one can find $n-1$ pairwise orthogonal Latin squares. For example, Figure 2.3 shows four pairwise orthogonal Latin squares of order five. If n is the power of a prime number, then one can construct such Latin squares using the finite field $\mathbb{F}_n$ with n elements. One such construction is stated in [Bose, 1938] and works as follows.

For every $a \in \mathbb{F}_n \setminus \{0\}$ we define a Latin square L_a by $L_a[x, y] = x + ay$, where the rows and columns are labeled (arbitrarily) with the elements from $\mathbb{F}_n$.

The Latin squares in Figure 2.3 are obtained in this way for $a = 1, 2, 3, 4$, where the columns and rows are labeled with $0, 1, 2, 3, 4$ consecutively.

Bose shows that L_a is indeed a Latin square and $L_a, L_{a'}$ are orthogonal for $a \neq a'$, $a, a' \in \mathbb{F}_n \setminus \{0\}$. We need another property of these Latin squares. We say that a pair of Latin squares L, L' intersects row-wise, if for all $i, i' \in \{1, \dots, n\}$ there exists an index $j \in \{1, \dots, n\}$ such that $L[i, j] = L'[i', j]$. Not all orthogonal Latin squares intersect row-wise but the ones obtained from Bose's construction do as an easy calculation shows.

Claim: The Latin square L_a and $L_{a'}$ constructed as above intersect row-wise for $a, a' \in \mathbb{F}_n \setminus \{0\}$ with $a \neq a'$.

Proof. Let $x, x' \in \mathbb{F}_n$ be arbitrary, and set $y = (x' - x)(a - a')^{-1}$. Then

$$x + ay - (x' + a'y) = x - x' + (a - a')y = x - x' + x' - x = 0$$

holds and this implies $L_a[x, y] = L_{a'}[x', y]$. □

Now, we have all ingredients to define for each $d = q^2 = p^{2k}$, where p is prime, a partitioned hypergraph H with $\nu_E(H) = 1$, maximum part size d, and vertex degree equal to q for all vertices.

The vertex set consists of $U := \{u_0, u_1, \dots, u_{d-1}\}$ and $W := \{w_0, w_1, \dots, w_{d-1}\}$. We divide each of the sets U and W into q subsets of size q, namely $W_i := \{w_{iq}, w_{iq+1}, \dots, w_{iq+q-1}\}$, and $U_i := \{u_{iq}, u_{iq+1}, \dots, u_{iq+q-1}\}$ for $i = 0, \dots, q-1$. Notice that the subsets W_i and U_i are only used to define the hyperedges of H and they are not all parts of H. Namely, the finest partition will by $W_0, W_1, \dots, W_q$ and U.

Next, we define the hyperedges of H. There are q types of them, where each type i consists of q different hyperedges containing W_i. Type 0 consists of the hyperedges $W_0 \cup U_i$ for $0 \leq i \leq q-1$. The $q-1$ other types are constructed by using the $q-1$ mutually orthogonal Latin squares of Bose's construction. Therefore, let $a_1, \dots, a_{q-1}$ be the non-zero elements of $\mathbb{F}_q$. For every row $x \in \mathbb{F}_q$ of the Latin square L_{a_i} we construct the hyperedge

$$e(i, x) := W_i \cup \{u_{jq+c} : c = L_{a_i}[x, a_j], 0 \leq j \leq q-1\}.$$

In this way, we get q hyperedges for every a_i with $i \geq 1$, which contain all vertices of W_i and have no vertices of U in common. Furthermore, $e(i, x)$ intersects every $U_{i'}$ for $i' \in \{0, \dots, q-1\}$ in exactly one vertex, thus $e(i, x)$ has a non-empty intersection with every hyperedge of the form $W_0 \cup U_{i'}$. Also, $e(i, x) \cap e(i', x') \neq \emptyset$ for $i \neq i'$, because L_{a_i} and $L_{a_{i'}}$ intersect row-wise.

Formally, the hypergraph H has vertex set $V(H) := U \cup W$ and hyperedge set $E(H) := \{W_0 \cup U_i : 0 \leq i \leq q-1\} \cup \{e(i,x) : 1 \leq i \leq q-1, x \in \mathbb{F}_q \setminus \{0\}\}$. The finest partition of H is $U, W_0, \dots, W_{q-1}$. All in all, we have constructed a partitioned hypergraph with maximum part size q^2, maximum degree q, whose hyperedges intersect pairwise. The maximum size of a matching is one whereas $x \equiv 1/q$ is a fractional matching of size $q = \sqrt{d}$. Thus, we get $\nu^*_E(H) \geq \sqrt{d}\nu_E(H)$ for every hypergraph H constructed in this way.

Example 2.17. We illustrate the construction defined above for $d = 9 = 3^2$. In this case, the hyperedges of type 0 are $\{w_0, w_1, w_2, u_0, u_1, u_2\}$, $\{w_0, w_1, w_2, u_3, u_4, u_5\}$ and $\{w_0, w_1, w_2, u_6, u_7, u_8\}$.

The other hyperedges are constructed using the following two Latin squares.

0	1	2
1	2	0
2	0	1

0	2	1
1	0	2
2	1	0

The first Latin square gives rise to the hyperedges of type 1, namely:

- $\{w_3, w_4, w_5, u_{0+0}, u_{3+1}, u_{6+2}\} = \{w_3, w_4, w_5, u_0, u_4, u_8\}$,
- $\{w_3, w_4, w_5, u_{0+1}, u_{3+2}, u_{6+0}\} = \{w_3, w_4, w_5, u_1, u_5, u_6\}$, and
- $\{w_3, w_4, w_5, u_{0+2}, u_{3+0}, u_{6+1}\} = \{w_3, w_4, w_5, u_2, u_3, u_7\}$.

The second one defines the hyperedges of type 2, which are

- $\{w_6, w_7, w_8, u_{0+0}, u_{3+2}, u_{6+1}\} = \{w_6, w_7, w_8, u_0, u_5, u_7\}$,
- $\{w_6, w_7, w_8, u_{0+1}, u_{3+0}, u_{6+2}\} = \{w_6, w_7, w_8, u_1, u_3, u_8\}$, and
- $\{w_6, w_7, w_8, u_{0+2}, u_{3+1}, u_{6+0}\} = \{w_6, w_7, w_8, u_2, u_4, u_6\}$.

We conclude this section by comparing Theorem 2.13 with the best approximation guarantees known for hypergraphs with bounded hyperedge size. A partitioned hypergraph with parts of size at most d can have hyperedges of size up to $2d$. This means that Theorem 2.1 would give a bound of $2d - 1 + 1/2d$ for the integrality gap of the fractional matching polytope, which is worse than $2\sqrt{d}$ (the bound obtained in Theorem 2.13). Furthermore, all approximation algorithms known have an approximation factor that is linear in the maximum size of a hyperedge and unless $\mathcal{P} = \mathcal{NP}$ there exist no algorithms with better guarantees, see Subsection 2.1.2.

On the other hand, given any r-uniform hypergraph H one can define a partitioned hypergraph H' with vertex set $U = V(H) \times \{0\}$, $W = V(H) \times \{1\}$, and hyperedges $\{\{(v,0),(v,1)\} : v \in e\}$ for all $e \in E(H)$. There is a one-to-one correspondence between matchings, as well as fractional matchings, in H and H'. Thus,

$\nu_b^*(H) = \nu_b^*(H')$ and $\nu_b(H) = \nu_b(H')$. We compare the bounds for the ratio of these numbers obtained by Theorem 2.13 with those of Theorem 2.1. In general it is possible that H' has only the trivial partition U, W, resulting in a factor of $2\sqrt{|V(H)|}$ for Theorem 2.13, which can be arbitrarily worse than $r(H) - 1 + 1/r(H)$. The construction of H' also works for non-uniform hypergraphs. Together with the $2 \cdot \sqrt{d}$-approximation algorithm we get a polynomial time $2 \cdot \sqrt{|V(H)|}$-approximation algorithm for the maximum weight matching problem on general hypergraphs. The approximation factor matches the one of the approximation algorithm given in [Halldórsson, 1999]. The $\Omega(|V(H)|^{0.5-\epsilon})$-hardness result for approximation algorithms for the maximum weight matching problem implies that the approximation factor in Theorem 2.13 cannot be improved substantially unless $\mathcal{P} = \mathcal{NP}$.

We conclude that neither of the Theorems 2.1, 2.13 dominates the other. Which of the both gives the better approximation factor highly depends on the concrete hypergraph structure.

2.3 Hall's and Kőnig's Theorem in Graphs and Hypergraphs

In this section we investigate the relationship between Kőnig's theorem, Hall's theorem, and a deficiency version of Hall's theorem in several graph and hypergraph classes. In the first subsection we consider graphs and in the second one we focus on hypergraphs.

2.3.1 The Graph Case

For a graph G we just write $\nu(G)$ instead of $\nu_E(G)$ for the maximum size of a matching in G. Similar, we drop E from the index of $\tau(G)$, $\tau^*(G)$, and $\nu^*(G)$. Graphs for which the vertex cover number $\tau(G)$ equals the matching number $\nu(G)$ are called *Kőnig-Egerváry graphs*. If $\tau(G)$ and $\tau^*(G)$ coincide, then $\tau(G) = \nu(G)$ follows, i.e., G is already a Kőnig-Egerváry graph. However, $\nu(G) = \nu^*(G)$ does not imply $\tau(G) = \nu(G)$. For example, the complete graph on four vertices K_4 satisfies $\nu(K_4) = \nu^*(K_4) = 2$ but $\tau(K_4) = 3$. Indeed, every graph with an even number of vertices and a perfect matching satisfies $\nu(G) = \nu^*(G) = |V(G)|/2$. In general, $\nu^*(G)$ is an upper bound on $\nu(G)$ and we are interested in graphs for which both values are equal.

Definition 2.18 (stable graph)**.** A graph G is *stable* if $\nu(G) = \nu^*(G)$.

The notion of a stable graph comes from the fact that a graph is stable if and only if the vertices that are missed by at least one maximum matching form a stable set, see for example [Deng et al., 1999].

To state a variant of Hall's theorem we need the following two notions, which are defined in [Levit and Mandrescu, 2012].

Definition 2.19 (deficiency and critical difference of a graph)**.** The *deficiency* $\operatorname{def}(G)$ of a graph G is the number of vertices not covered by a maximum matching, i.e., $\operatorname{def}(G) := |V(G)| - 2\nu(G)$. The *critical difference* of a graph G is defined to be $d(G) := \max\{|S| - |N(S)| : S \text{ is a stable set}\}$, where $N(S)$ is the set of neighbors of S in G.

If S is a stable set, then every matching matches a vertex from S to one of $N(S)$. In particular, if $|S| > |N(S)|$, then every matching misses at least $|S| - |N(S)|$ vertices of S. This implies that the critical difference gives a lower bound on the deficiency of a graph. We introduce a notion for graphs where this bound is tight.

Definition 2.20 (strong Hall property)**.** A graph G has the *strong Hall property* if its deficiency is equal to its critical difference, i.e., if $\operatorname{def}(G) = d(G)$.

Every Kőnig-Egerváry graph has the strong Hall property, which is for example proven in [Levit and Mandrescu, 2012]. We show that there exist also other graphs with the strong Hall property. Furthermore, we consider the class of graphs for which Hall's theorem is true.

Definition 2.21 (Hall property)**.** A graph has the *Hall property* if it has a perfect matching or a stable set S with less than $|S|$ neighbors.

Observe that a graph with a perfect matching cannot have a stable set S with less than $|S|$ neighbors and a graph with such a set S cannot have a perfect matching. Thus, the two statements in the definition of the Hall property are mutually exclusive.

For graphs the deficiency version of Hall's theorem seems to be an easy reformulation of Kőnig's theorem. So, one might expect, that the graphs with the strong Hall property are exactly the Kőnig-Egerváry graphs. However, this is not the case. Namely, it is enough that the maximum size of a matching equals the maximum size of a fractional matching as we show in the following theorem.

Theorem 2.22. *A graph has the strong Hall property if and only if it is stable.*

Proof. We show that for a graph G the statements $\operatorname{def}(G) = d(G)$ is equivalent to $\nu(G) = \nu^*(G)$ using integer programming methods and the well-known fact that the fractional vertex cover polyhedron is half-integral, see for example Theorem 7.5.3 in [Lovász and Plummer, 1986].

The critical difference of a graph G can be computed via the following integer program.

$$\max \textstyle\sum_{v \in V(G)} (b_v - r_v) \tag{2.1}$$

$$s.t. \textstyle\sum_{v \in e} (r_v - b_v) \geq 0 \quad \forall e \in E(G) \tag{2.2}$$

$$b_v, r_v \in \{0,1\} \qquad \forall v \in V(G). \tag{2.3}$$

Indeed, if S is a stable set, then

$$b_v := \begin{cases} 1, & v \in S \\ 0, & v \notin S \end{cases}, \; r_v := \begin{cases} 1, & v \in N(S) \\ 0, & v \notin N(S) \end{cases}$$

satisfies the inequalities of type (2.2) and its objective value is $|S| - |N(S)|$. Thus, the optimal value of (2.1)-(2.3) is at least $d(G)$.

On the other hand, let b^*, r^* be an optimal solution to (2.1)-(2.3). If there exists a vertex v with $b^*_v = r^*_v = 1$, then decreasing both variables to zero does not affect the feasibility of r^*, b^* and it does not change the objective value. Thus, for every vertex v we can assume that at most one of r^*_v, b^*_v is non-zero. We set $S := \{v\colon b^*_v = 1\}$ and $T := \{v\colon r^*_v = 1\}$. Let $\{v, w\}$ be an edge with $v \in S$, i.e., $b^*_v = 1$. By the modification of r^*, b^* we have $r^*_v = 0$, and inequality (2.2) for $e = \{v, w\}$ states that $(0 - 1) + (r^*_w - b^*_w) \geq 0$. Thus, we get $r^*_w - b^*_w \geq 1$, which implies $r^*_w = 1$. By the definition of T, $r^*_w = 1$ means $w \in T$. We have shown that $T \supseteq N(S)$ and S is a stable set. The objective value of r^*, b^* is equal to $|S| - |T| \leq |S| - |N(S)| \leq d(G)$. In total, the optimal value of (2.1)-(2.3) is equal to the critical difference of G.

Now, we reformulate the integer program (2.1)-(2.3) using the variable transformation $w_v := \frac{r_v - b_v + 1}{2}$ for all $v \in V(G)$. We obtain the half-integer program

$$\max \left(|V(G)| - 2 \cdot \textstyle\sum_{v \in V(G)} w_v \right) \tag{2.4}$$

$$s.t. \textstyle\sum_{v \in e} w_v \geq 1 \qquad \forall e \in E(G) \tag{2.5}$$

$$w_v \in \{0, \tfrac{1}{2}, 1\} \qquad \forall v \in V(G), \tag{2.6}$$

which has the same optimal value as the linear program

$$|V(G)| - \quad 2 \cdot \min \textstyle\sum_{v \in V(G)} w_v \tag{2.7}$$

$$s.t. \textstyle\sum_{v \in e} w_v \geq 1 \qquad \forall e \in E(G) \tag{2.8}$$

$$0 \leq w_v \leq 1 \qquad \forall v \in V(G) \tag{2.9}$$

because (2.7)-(2.9) has always an optimal solution with entries in $\{0, \frac{1}{2}, 1\}^{V(G)}$ by Theorem 7.5.3 in [Lovász and Plummer, 1986].

On the other hand, the deficiency of G is equal to the optimal value of the following integer program:

$$|V| - \quad 2 \cdot \max \textstyle\sum_{e \in E(G)} x_e \tag{2.10}$$

$$\textstyle\sum_{e \in \delta(v)} x_e \leq 1 \qquad \forall v \in V(G) \tag{2.11}$$

$$x_e \in \{0, 1\} \qquad \forall e \in E(G). \tag{2.12}$$

This implies that $\operatorname{def}(G) = d(G)$ if and only if

$$\begin{array}{lllll}
\min \sum\limits_{v \in V(G)} w_v & & = & \max \sum\limits_{e \in E(G)} x_e & \\
\sum\limits_{v \in e} w_v \geq 1 & \forall e \in E(G) & & \sum\limits_{e \in \delta(v)} x_e \leq 1 & \forall v \in V(G) \\
0 \leq w_v \leq 1 & \forall v \in V(G) & & x_e \in \{0,1\} & \forall e \in E(G).
\end{array}$$

The program on the right is an integer programming formulation of the maximum matching problem. The linear program obtained by relaxing the integrality constraints $x_e \in \{0,1\}$ by $0 \leq x_e \leq 1$ for all $e \in E(G)$ is the dual of the linear program on the left. It follows that $\mathrm{def}(G) = d(G)$ if and only if $\nu(G) = \nu^*(G)$, i.e., G is a stable graph. □

Next, we characterize graphs with the Hall property. A *fractional perfect matching* in a graph G is a fractional matching $x : E(G) \to \mathbb{Q}_{\geq 0}$ with $x(\delta(v)) = 1$ for all $v \in V(G)$. It is well known that a graph has a fractional perfect matching if and only if $|N(S)| \geq |S|$ for all stable sets S holds, see for example Corollary 6.1.5 in [Lovász and Plummer, 1986]. Liu and Liu give another characterization of those graphs that have a fractional perfect matching, see [Liu and Liu, 2002]. Their result uses the so-called Gallai-Edmonds decomposition of a graph. Recall, that the *Gallai-Edmonds decomposition* of a graph G is a partition of its vertex set into three sets $D(G), A(G), C(G)$ with the property that $D(G)$ is the set of vertices missed by at least one maximum matching, $A(G)$ is the set of vertices in $V(G) \setminus D(G)$ adjacent to $D(G)$, and $C(G)$ contains all remaining vertices.

The Gallai-Edmonds decomposition tells a lot about the structure of maximum matchings in a graph. It also gives some information about fractional matchings as the following result of Liu and Liu shows.

Corollary 2.23. *[Liu and Liu, 2002] A graph G has a perfect fractional matching if and only if it has a matching covering every trivial component (a component consisting of exactly one vertex) of $G[D(G)]$.*

Using Corollary 2.23 we characterize graphs with the Hall property.

Corollary 2.24. *For a given graph G, let $D(G), A(G), C(G)$ be its Gallai-Edmonds decomposition, and $D'(G)$ the vertices of $D(G)$ that form a connected component in $G[D(G)]$ consisting just of one vertex and no edge.*

The graph G has the Hall property if and only if

(a) $D(G) = \emptyset$, which means that G has a perfect matching, or

(b) the subgraph of G induced by $A(G)$ and $D(G)$ has no matching covering $D'(G)$.

Proof. First, suppose G has the Hall property and G has no perfect matching. This means that there exists a stable set S of G with $|S| > |N(S)|$, which implies that G has no perfect fractional matching. By Corollary 2.23, G has no matching covering $D'(G)$, which is equivalent to (b) by the construction of $A(G)$ and $D(G)$.

On the other hand, if G has a perfect matching, then G has clearly the Hall property. In case (b), using again Corollary 2.23, we get that G has no perfect fractional matching. Thus, there exists a stable set S of G with $|S| > |N(S)|$. □

This characterization shows again the strength of the Gallai-Edmonds decomposition. Namely, it can be used to characterize Kőnig-Egerváry graphs, as well as graphs with the strong Hall and the Hall property. In the first two cases, the set $D(G)$ of vertices missed by any maximum size matching has to be a stable set in G, whereas in the latter case it is possible that the subgraph of G induced by $D(G)$ contains non-trivial components, however the trivial components must have some special structure in G.

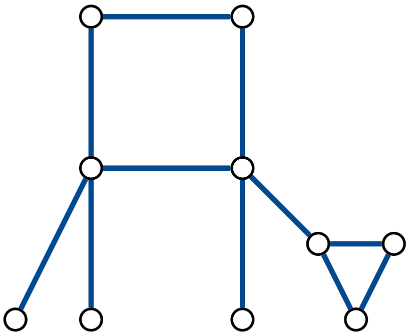


Figure 2.4: A semi-bipartite graph that is not stable.

Example 2.25. A class of graphs that are not necessarily stable but have the Hall property are semi-bipartite graphs with an even number of vertices, where a graph is *semi-bipartite* if every two vertex disjoint odd cycles are connected by an edge. In particular, every graph with at most one odd cycle is semi-bipartite, see Figure 2.4 for an example.

Fulkerson, Hoffman, and McAndrew show in [Fulkerson et al., 1965] that every semi-bipartite graph with an even number of vertices has the Hall property. This result also follows from Theorem 2.22 and Corollary 2.24. Namely, if G is semi-bipartite and stable, then it has even the strong Hall property. If G is a semi-bipartite, non-stable graph, then either G has a perfect matching or $D(G)$ is non-empty. In the latter case, $G[D(G)]$ contains exactly one non-trivial connected component. Otherwise, $G[D(G)]$ contains two factor-critical connected components (see

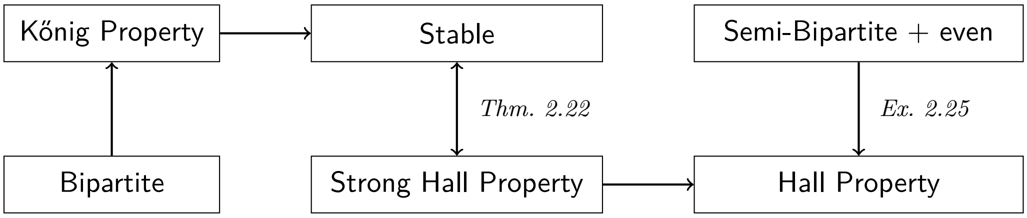


Figure 2.5: Summary of the relations between the investigated graph properties.

Theorem 3.2.1 in [Lovász and Plummer, 1986]). Both of these components are non-bipartite and contain an odd cycle. Two odd cycles obtained in this way are vertex disjoint and not connected by an edge in G, which is impossible by the definition of a semi-bipartite graph. Thus, $G[D(G)]$ has exactly one non-trivial component. If G has an even number of vertices, then by parity arguments $G[D(G)]$ must contain at least $|A(G)| + 2$ components. Thus, $|D'(G)| = |D(G)| - 1 \geq |A(G)| + 1$, and we are in case (b) of Corollary 2.24. If G has an odd number of vertices, then it is possible that $|D'(G)| = |A(G)|$ and there exists a perfect matching between $D'(G)$ and $A(G)$, i.e., a semi-bipartite graph with an odd number of vertices might not have the Hall property.

We conclude this subsection by discussing the relationship between different properties and graph classes that we investigated before. Table 2.5 gives a schematic overview where an arrow from one property to another indicates an implication, and arrows implied by transitivity are not drawn for better readability. For example, a bipartite graph is a Kőnig-Egerváry graph (by Kőnig's theorem), it is stable, has the strong Hall, and the Hall property. A bipartite graph is also semi-bipartite, however, we only considered semi-bipartite graphs with an even number of vertices. Thus, there is no arrow from "Bipartite" to "Semi-Bipartite + even". We argue why exactly the depicted implications hold.

For "Bipartite" this was already done above. Every Kőnig-Egerváry graph is stable by the characterization of Kőnig-Egerváry graphs in [Lovász, 1983]. Lovász proves that a graph G is Kőnig-Egerváry if and only if $D(G)$ is a stable set in G and $G[V(G) \setminus (D(G) \cup N(D(G)))]$ is Kőnig-Egerváry. For a graph G to be stable $D(G)$ only needs to be a stable set in G.

The equivalence of "Stable" and "Strong Hall Property" was shown in Theorem 2.22. By definition, the "Strong Hall Property" implies the "Hall Property", and semi-bipartite graphs with an even number of vertices are an example that the reverse implication is false. Finally, a semi-bipartite graph with an even number of vertices has the Hall property by Corollary 2.24 but it is in general neither bipartite, nor Kőnig-Egerváry, nor stable as the graph depicted in Figure 2.4 shows.

2.3.2 The Hypergraph Case

In this subsection we regard generalizations of Hall's Theorem to some special hypergraphs, in particular normal hypergraphs. Conforti, Cornuéjols, Kapoor, and Vušković give with Theorem 2.8 ([Conforti et al., 1996]) one possibility for a Hall-type theorem for hypergraphs. Their result motivates the following definition.

Definition 2.26 (Hall property)**.** We say that a hypergraph has the *Hall property* if H has a perfect matching or there exists a pair (R, B) of disjoint vertex sets $R, B \subseteq V(H)$ with $|e \cap R| \geq |e \cap B|$ for all hyperedges e and $|R| < |B|$.

Observe that the two properties of this definition are mutually exclusive. If H has a perfect matching M and (R, B) is a pair of vertex sets with $|e \cap R| \geq |e \cap B|$ for all $e \in E(H)$, then

$$|R| = \sum_{e \in M} |e \cap R| \geq \sum_{e \in M} |e \cap B| = |B|.$$

Using Definition 2.26, Theorem 2.8 can be restated as "Every balanced hypergraph has the Hall property". We want to find other classes of hypergraphs with the Hall property. One starting point is to look at hypergraphs satisfying Kőnig's theorem.

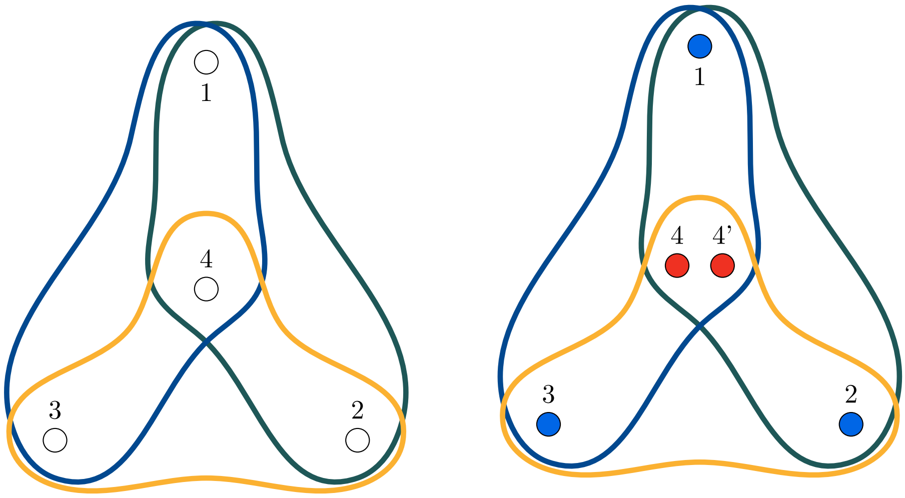
Definition 2.27 (Kőnig property)**.** A hypergraph H has the *Kőnig property* if the maximum size of a matching equals the minimum size of a vertex cover, i.e., $\nu_E(H) = \tau_E(H)$.

As every Kőnig-Egervary graph has the Hall property, one might expect that the Kőnig property implies the Hall property for hypergraphs. This is not the case as the following counterexample shows. The constructed hypergraph is normal, and thus has the Kőnig property.

Example 2.28. Let H be the hypergraph on the vertex set $\{1, 2, 3, 4\}$ with hyperedges $\{1, 2, 4\}$, $\{2, 3, 4\}$, $\{1, 3, 4\}$, see Figure 2.6a. This is the smallest normal hypergraph that is not balanced. We show that H does not have the Hall property.

As all hyperedges intersect in vertex 4 and there is no hyperedge containing all vertices, H has no perfect matching. If (R, B) is a pair of disjoint vertex sets with $|e \cap R| \geq |e \cap B|$ for all $e \in E(H)$, then we claim that $|R| \geq |B|$. Indeed, if $|B| \geq 2$, then there exists a hyperedge with $|e \cap B| \geq 2$ and $|e \cap R| \leq 1$ as every pair of vertices of H is contained in a hyperedge of size three. This shows that $|B| \leq 1$. Clearly, if $|B| = 1$, then $|R| \geq 1$. In total, we get $|R| \geq |B|$.

The hypergraph H' obtained from H by adding a copy of vertex 4, that is, the hypergraph on $\{1, 2, 3, 4, 4'\}$ with hyperedges $\{1, 2, 4, 4'\}$, $\{2, 3, 4, 4'\}$, $\{1, 3, 4, 4'\}$ (see Figure 2.6b), has the Hall property. Namely, if we choose $B := \{1, 2, 3\}$ and $R := \{4, 4'\}$, then $|e \cap R| = 2 = |e \cap B|$ for all $e \in E(H')$ and $|R| < |B|$ holds.



(a) A normal hypergraph without the Hall property.

(b) The hypergraph on the left after duplicating vertex 4 has the Hall property.

Figure 2.6: Does Theorem 2.8 hold for normal hypergraphs?

In the remainder of this subsection we show that the multiplication trick of the previous example always works and give a generalization of Hall's theorem to normal hypergraphs including a deficiency variant.

First, we look at the dual hypergraphs of normal ones. These hypergraphs are called *perfect* by Schrijver, see Chapter 82 in [Schrijver, 2002]. The name perfect comes from a characterization via perfect graphs that we will use as a definition.

Definition 2.29 (perfect hypergraph)**.** A hypergraph H is *perfect* if there exists a perfect graph G on the same vertex set such that the maximal hyperedges of H correspond one-to-one to the maximal cliques of G.

Every balanced hypergraph is perfect. This follows from the fact that every balanced hypergraph is normal and the observation that the dual of a balanced hypergraph is balanced.

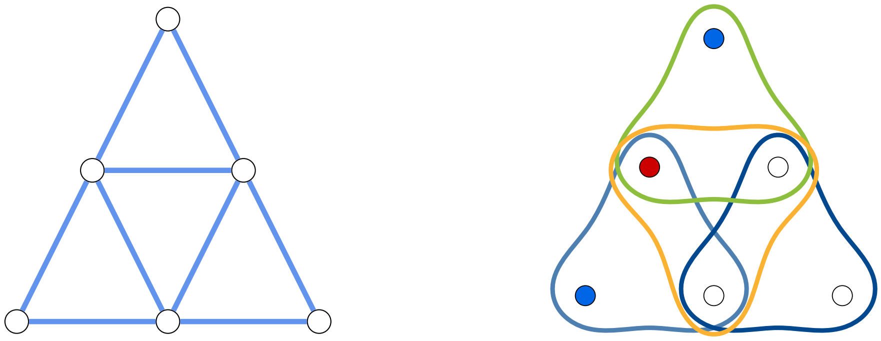
We prove that uniform perfect hypergraphs have the Hall property. This is particularly interesting as perfect hypergraphs do not have the Kőnig property in general. For example, the maximum size of a matching in the hypergraph depicted in Figure 2.7b is one whereas every E-vertex cover has size at least two.

Theorem 2.30. *Every uniform perfect hypergraph has the Hall property.*

Proof. Let H be a uniform perfect hypergraph without a perfect matching. As H is a perfect hypergraph, the graph $G(H) := (V(H), \{\{v, w\} : \exists e \in E(H) \text{ with } v, w \in e\})$

is perfect, and the maximal cliques of $G(H)$ correspond to the hyperedges of H (see Theorem 82.4 in [Schrijver, 2002]). As H is uniform, every maximal clique of $G(H)$ has size r, where r is the rank of H. This implies that there exists a stable set S of size at most $|V(H)|/r$ that intersects every maximal clique of $G(H)$(for example let S be the smallest color class of an r-coloring of $G(H)$).

The minimum size of a clique cover of $G(H)$ must be larger than $|V(H)|/r$ because H has no perfect matching and it is r-uniform. As $G(H)$ is perfect, there exists a maximal stable set $\tilde{S}$ of size greater than $|V(H)|/r$. We set $R := S \backslash \tilde{S}$ and $B := \tilde{S} \backslash S$. Observe that both sets are non-empty as S and $\tilde{S}$ are maximal stable sets of different sizes. By definition, $R \cap B = \emptyset$, and $|R| = |S| - |S \cap \tilde{S}| < |\tilde{S}| - |S \cap \tilde{S}| = |B|$. Furthermore, $|Q \cap B| \leq 1$ for all maximal cliques Q of $G(H)$ and if $|Q \cap B| = 1$, then $1 = |Q \cap (\tilde{S} \setminus S)| = |Q \cap \tilde{S}| - |Q \cap (\tilde{S} \cap S)|$, and thus $Q \cap (\tilde{S} \cap S) = \emptyset$. In particular, we get $|Q \cap R| = |Q \cap S| = 1$ for all maximal cliques Q, which shows that $|e \cap R| \geq |e \cap B|$ for all $e \in E(H)$. □



(a) The graph $G(H)$.

(b) A 3-uniform perfect hypergraph H.

Figure 2.7: A perfect graph and the hypergraph with its maximal cliques as hyperedges.

Non-uniform perfect hypergraphs do not have the Hall property in general, as the following example shows:

Example 2.31. Consider the hypergraph H on the vertex set $\{1, 2, 3, 4\}$ with hyperedges $\{1, 2, 4\}$, $\{2, 3, 4\}$, $\{2, 3\}$, $\{3, 4\}$, and $\{2, 4\}$ depicted in Figure 2.8. This hypergraph is perfect because the maximal edges $\{1, 2, 4\}, \{2, 3, 4\}$ correspond to the maximal cliques in the perfect graph

$$G = (\{1, 2, 3, 4\}, \{\{1, 2\}, \{1, 4\}, \{2, 3\}, \{2, 4\}, \{3, 4\}\}).$$

H has no perfect matching, however, $|R| \geq |B|$ holds for all disjoint vertex sets $R, B \subseteq V(H)$ with $|e \cap R| \geq |e \cap B|$ for all $e \in E(H)$. Namely, $|e \cap R| \geq |e \cap B|$

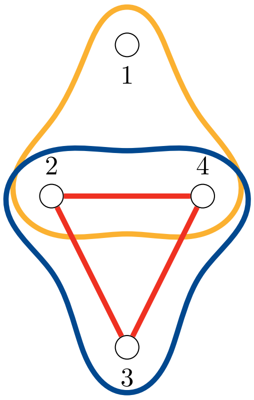


Figure 2.8: An example showing that Theorem 2.30 does not hold for non-uniform perfect hypergraphs.

for $e = \{2,3,4\}$ implies that $|B \cap \{2,3,4\}| \leq 1$. Thus, $|B| \leq 2$. If $|B| = 1$, then clearly $|R| \geq 1$. If $|B| = 2$, then $B = \{1, i\}$ for some $i \in \{2,3,4\}$. In this case, $|\{1,2,4\} \cap R| \geq |\{1,2,4\} \cap B|$ implies $B = \{1,3\}$. As $|e \cap R| \geq |e \cap B|$ for $e = \{2,3\}$, and $\{3,4\}$, we get $R = \{2,4\}$, and thus $|R| = 2 = |B|$. Therefore, H has not the Hall property.

In the remainder of this subsection, we give a deficiency variant of Theorem 2.8 for normal hypergraphs. Analogous to graphs, we define the *deficiency* of a hypergraph,

$$\operatorname{def}(H) := \min\{|V(H)| - |V(M)| : \text{ M is a matching of } H\},$$

to be the minimum number of vertices that are exposed by a matching.

In the same vein, we define the *critical difference* of a hypergraph H by

$$d(H) := \max\{|B| - |R| : R, B \subseteq V(H), R \cap B = \emptyset, |e \cap R| \geq |e \cap B| \forall e \in E(H)\},$$

and a pair (R, B) of vertices attaining the maximum at the right hand side of this definition a *critical pair*. This generalizes the definition of the critical difference in graphs by [Levit and Mandrescu, 2012].

As in the graph case, the critical difference of a hypergraph gives a lower bound on its deficiency. Indeed, if M is a matching covering as many vertices as possible in a hypergraph H, and (R, B) is a critical pair of H, then

$$\begin{aligned} d(H) = |B| - |R| &= \sum_{e \in M} |e \cap B| + |B \setminus V(M)| - \left(\sum_{e \in M} |e \cap R| + |R \setminus V(M)| \right) \\ &\leq |B \setminus V(M)| - |R \setminus V(M)| \leq |B \setminus V(M)| \leq |V(H) \setminus V(M)| = \operatorname{def}(H). \end{aligned}$$

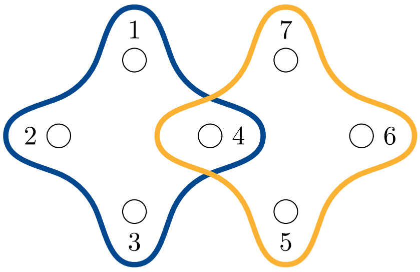


Figure 2.9: A balanced hypergraph with $\mathrm{def}(H) = 3 > 1 = d(H)$.

However, even for a balanced hypergraph the critical difference can be smaller than the deficiency. In [Huck and Triesch, 2002] the following example is given, which shows that the gap between the critical difference and the deficiency can be arbitrarily large even for very simple balanced hypergraphs.

Example 2.32. Let H be the hypergraph on the vertices $1, 2, \ldots, 2n + 1$ having the two hyperedges $e_1 = \{1, \ldots, n + 1\}$, $e_2 = \{n + 1, \ldots, 2n + 1\}$, see Figure 2.9 for a picture of the case $n = 3$. As H has only two hyperedges it is obviously balanced. Furthermore, every non-empty matching misses n vertices, so $\mathrm{def}(H) = n$. However, $d(H) = 1$ as there is no pair $R, B \subseteq V$ with $|e_i \cap B| \leq |e_i \cap R|$ for $i = 1, 2$ and $|B| - |R| > 1$, and thus $R = \{n + 1\}$, $B = \{n, n + 2\}$ is a critical pair.

If we could take n copies of the vertex $n + 1$ into the set R and all other vertices into B, then we would get a pair R, B with $|e_i \cap R| = n = |e_i \cap B|$ $(i = 1, 2)$ and $|B| - |R| = 2n - n = n$. This means that the deficiency of H equals the critical difference of the hypergraph in which vertex $n + 1$ is “multiplied” n times.

We show that a deficiency variant of Hall’s Theorem can be derived using a multiplication trick as in the Examples 2.28, and 2.32. This trick can be formalized as follows.

Definition 2.33 (Vertex multiplication, [Berge, 1984])**.** Let H be a hypergraph, $v \in V(H)$ be a fixed vertex, and $\lambda \in \mathbb{N}$. The hypergraph obtained by *multiplying v by λ* is the hypergraph that arises from H by replacing the vertex v by λ new vertices $(v, 1), \ldots, (v, \lambda)$ and every hyperedge e containing v by the new hyperedge $e \setminus \{v\} \cup \{(v, 1), \ldots, (v, \lambda)\}$.

For $c \in \mathbb{N}^{V(H)}$, $H^{(c)}$ is the hypergraph obtained from H by multiplying each vertex v by c_v, and $H^{(c)}$ is called a *multiplication* of H. For every $e \in E(H)$ we denote the corresponding hyperedge in $E(H^{(c)})$ by $e^{(c)}$. If all entries of c are equal to some constant $k \in \mathbb{N}$, we also write $H^{(k)}$ and $e^{(k)}$.

Informally speaking, multiplying a vertex by some number λ means replacing this vertex by λ indistinguishable copies. Observe that the vertex multiplication operation is different form the vertex expansion defined in Definition 1.28.

Multiplying a vertex by some λ does not change the intersection behavior of the hyperedges as $\lambda \geq 1$. Thus, the set of matchings in H corresponds one-to-one to the set of matchings in $H^{(c)}$. Similarly, there is a one-to-one correspondence between the set of perfect matchings in H and those in $H^{(c)}$. The maximum degree also stays unchanged. This implies that if H is normal, then $H^{(c)}$ is also normal.

One problem arises when looking at the critical difference of the multiplied hypergraph $H^{(k)}$ for $k \in \mathbb{N}$. If R, B is a critical pair of $H^{(k)}$ with $d(H^{(k)}) = |B| - |R| > 0$, then we can define a pair of disjoint vertex sets R', B' in $H^{(lk)}$ by taking l times the number of copies of $v \in V(H)$ that R or B contains. For all $e^{(lk)} \in E^{(lk)}$ we have $|e^{(lk)} \cap R'| = l \cdot |e^{(k)} \cap R| \geq l \cdot |e^{(k)} \cap B| = |e^{(lk)} \cap B'|$. It holds that $|B'| - |R'| = l \cdot (|B| - |R|)$, thus $d(H^{(lk)}) \geq l \cdot d(H^{(k)})$. This implies that $d(H^{(k)})$ is not a lower bound on $\operatorname{def}(H)$ for all $k \in \mathbb{N}$. We can overcome this problem by considering a restricted version of the critical difference, where we allow to take multiple copies of a vertex into R but only one copy of each vertex into B.

Definition 2.34 (multiplied critical difference)**.** For a multiplication $H^{(k)}$ of a hypergraph H we define its *multiplied critical difference* by

$$\begin{aligned} d^*(H^{(k)}) := \max\{|B| - |R| : {} & R, B \subseteq V(H^{(k)}),\ R \cap B = \emptyset, \\ & |e \cap R| \geq |e \cap B| \ \forall e \in E(H^{(k)}), \\ & |B \cap \{(v,1),(v,2),\dots,(v,k)\}| \leq 1 \ \forall v \in V(H)\}. \end{aligned} \tag{2.13}$$

By definition, we have $d(H^{(k)}) \geq d^*(H^{(k)})$ for all $k \in \mathbb{N}$. Furthermore, we observed that $\operatorname{def}(H) \geq d(H)$ but $d(H^{(k)}) > \operatorname{def}(H)$ for large values of $k \in \mathbb{N}$. We show that $\operatorname{def}(H) \geq d^*(H^{(k)})$ for all $k \in \mathbb{N}$. In other words, the multiplied critical difference always gives a lower bound on the deficiency.

Observation 2.35. *Let H be a hypergraph and k some natural number. The deficiency of H is greater or equal to the multiplied critical difference of $H^{(k)}$.*

Proof. Let M be a matching covering as many vertices as possible, and let $R, B \subseteq V(H^{(k)})$ be a pair attaining the maximum in (2.13). The matching M corresponds to a matching $M^{(k)} := \{e^{(k)} | e \in M\} \subseteq E(H^{(k)})$ of $H^{(k)}$ covering the vertices $C := V(M^{(k)}) \subseteq V(H^{(k)})$.

Counting R and B hyperedge-wise and using $|e \cap R| \geq |e \cap B|$ for all $e \in E(H^{(k)})$ gives

$$\begin{aligned} d^*(H^{(k)}) &= |B| - |R| \\ &= \left(\sum_{e^{(k)} \in M^{(k)}} |e^{(k)} \cap B| \right) + |B \setminus C| - \left(\sum_{e^{(k)} \in M^{(k)}} |e^{(k)} \cap R| \right) - |R \setminus C| \\ &\leq |B \setminus C| - |R \setminus C| \leq |B \setminus C| \\ &\leq |V(H) \setminus V(M)| = \operatorname{def}(H). \end{aligned}$$

The last inequality holds because B contains at most one copy of each vertex. □

By Observation 2.35, $d^*(H^{(k)})$ is bounded for $k \to \infty$. This, together with $d^*(H^{(k)}) \geq d^*(H^{(k')})$ for $k \geq k'$, implies that $d^*(H^{(k)})$ converges. The next lemma shows that if H is a normal hypergraph, then the limit of $d^*(H^{(k)})$ for $k \to \infty$ is equal to the deficiency for a normal hypergraph H. In short, we obtain a deficiency version of Hall's Theorem for normal hypergraphs.

Lemma 2.36. *If H is a normal hypergraph of rank r, then def(H) $= d^*(H^{(r-1)})$.*

Proof. By Observation 2.35, it remains to show that $\operatorname{def}(H) \leq d^*(H^{(r-1)})$.

Let M be a V-maximum matching of H and x^* be a minimum size V-vertex cover of H. A normal hypergraph satisfies $\nu_V(H) = \tau_V(H)$, and this implies that $|V(M)| = \sum_{v \in V(H)} x^*_v$. As x^* is of minimum size, we have $x^*_v \leq \max_{e \in E(H)} |e| = r$ for all $v \in V(H)$. Thus, we can use x^* to define a pair (R, B) of disjoint vertex sets in $V(H^{(r-1)})$ as follows:

$$\begin{aligned} B &:= \{(v,1) | x^*_v = 0\}, \\ R &:= \{(v,i) | x^*_v \geq 2,\ 1 \leq i \leq x^*_v - 1\}. \end{aligned}$$

For every hyperedge $e \in E(H^{(r-1)})$ we get

$$|e \cap R| - |e \cap B| = \sum_{v \in e} (x^*_v - 1) \geq |e| - |e| = 0.$$

Furthermore,

$$|B| - |R| = \sum_{v \in V(H)} (1 - x^*_v) = |V(H)| - \sum_{v \in V(H)} x^*_v = |V(H)| - |V(M)| = \operatorname{def}(H),$$

which implies $\operatorname{def}(H) \leq d^*(H^{(r-1)})$. □

Lemma 2.36 gives the following combinatorial characterization for the (non-) existence of a perfect matching in a normal hypergraph.

Theorem 2.37 ([Beckenbach and Borndörfer, 2018])**.** *A normal hypergraph H of rank r has no perfect matching if and only if $H^{(r-1)}$ has a pair $R, B \subseteq V(H^{(r-1)})$ of disjoint vertex sets such that $|e \cap R| \geq |e \cap B|$ for all $e \in E(H^{(r-1)})$ and $|R| < |B|$, i.e., def(H) > 0 if and only if $d(H^{(r-1)}) > 0$.*

Proof. If $\operatorname{def}(H) > 0$, then $d^*(H^{(r-1)}) > 0$ by Lemma 2.36. As $d(H^{(r-1)}) \geq d^*(H^{(r-1)})$, it follows that $d(H^{r-1}) > 0$. On the other hand, if $d(H^{(r-1)}) > 0$, then $\operatorname{def}(H^{(r-1)}) > 0$, and thus also $\operatorname{def}(H) > 0$. □

The bound in Lemma 2.36 and Theorem 2.37 on the vertex multiplication factor is best possible as the following example shows.

Example 2.38. For every natural number $n \geq 3$, let H_n be the hypergraph with vertex set $\{1, \ldots, n, n+1\}$ and hyperedges $S \cup \{n+1\}$ for every subset S of $\{1, \ldots, n\}$ of size $n-1$ (see Figure 2.6a for H_3). Every two hyperedges of H_n intersect in vertex $n+1$, so H_n is a normal hypergraph without a perfect matching of rank n.

We claim that $N := n-1$ is the smallest natural number such that $H_n^{(N)}$ contains a pair R, B as in Theorem 2.37. Indeed, this can be shown by a simple calculation. Let $N \in \mathbb{N}$ such that there exists a pair $R, B \subseteq V(H_n^{(N)})$ with $|e \cap R| \geq |e \cap B|$ for all $e \in E(H_n^{(N)})$ and $|R| < |B|$.

For every $i \in V(H_n) = \{1, \ldots, n+1\}$ we define

$$y_i := |\{\text{copies of } i \text{ in } R\}| - |\{\text{copies of } i \text{ in } B\}|.$$

Every hyperedge of H_n is of the form $e = \{1, \ldots, n+1\} \setminus \{i\}$ for some $i \in \{1, \ldots, n\}$. As $|e^{(N)} \cap R| \geq |e^{(N)} \cap B|$, we get

$$y_1 + y_2 + \ldots + y_{n+1} - y_i \geq 0 \tag{2.14}$$

for all $i = 1, \ldots, n$. On the other hand, $\sum_{i=1}^{n+1} y_i = |R| - |B| < 0$, thus

$$y_i \leq \sum_{i=1}^{n+1} y_i < 0$$

holds for all $i = 1, \ldots, n$. The integrality of y_i implies $y_i \leq -1$ for $i = 1, \ldots, n$. This together with inequality (2.14) for $i = n$ gives

$$y_{n+1} \geq -y_1 - y_2 - \ldots - y_{n-1} \geq n-1.$$

It follows that $N \geq n-1$. □

2.3.3 Relation between Hypergraph Properties

In this subsection we give an overview of the relation between Kőnig's theorem and the different variants of Hall's theorem considered previously.

We define two Hall-type properties besides the one of Definition 2.26.

Definition 2.39 ((strong) multiplied Hall property)**.** A hypergraph H has the *multiplied Hall property* if it has a perfect matching or there exists a number $k \in \mathbb{N}$ such that $d(H^{(k)}) > 0$. If there exists $k \in \mathbb{N}$ such that $\operatorname{def}(H) = d^*(H^k)$, then H has the *strong multiplied Hall property*.

The strong multiplied Hall property implies the multiplied Hall property. Namely, if H has no perfect matching and $d^*(H^{(k)}) = \operatorname{def}(H)$, then $d(H^{(k)}) \geq d^*(H^{(k)}) > 0$.

In contrast to graphs, the Kőnig property does not imply the Hall property. For example every normal hypergraph has the Kőnig property, but there are normal

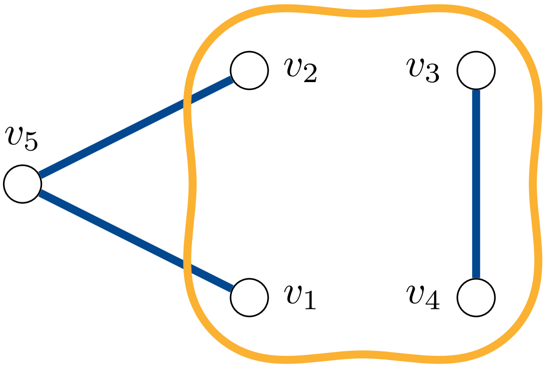


Figure 2.10: A hypergraph with the Kőnig property and without the multiplied Hall property.

hypergraphs without the Hall property, see Figure 2.6a. However, every normal hypergraph has the multiplied Hall property. Thus, the question arises whether every hypergraph with the Kőnig property has the multiplied Hall property. This is not the case as the following counterexample shows.

Example 2.40. We consider the hypergraph H on the vertex set $\{v_1, \dots, v_5\}$ with hyperedges $\{v_1, v_2, v_3, v_4\}$, $\{v_1, v_5\}$, $\{v_2, v_5\}$, $\{v_3, v_4\}$, see Figure 2.10. The maximum size of a matching of H is two (e.g. $\{\{v_1, v_5\}, \{v_3, v_4\}\}$) and the minimum size of a vertex cover is two (e.g. $\{v_5, v_3\}$). Thus, H has the Hall property. We claim that H does not have the multiplied Hall property.

Suppose to the contrary that there exists a number $k \in \mathbb{N}$ such that $d(H^{(k)}) > 0$, and let (R, B) be a critical pair of $H^{(k)}$. For every $i \in [5]$ we define

$$y_i := |\text{number of copies of } v_i \text{ in } R| - |\text{number of copies of } v_i \text{ in } B|.$$

As $|R| < |B|$ we get

$$y_1 + y_2 + y_3 + y_4 + y_5 < 0. \tag{2.15}$$

The inequalities $|e^{(k)} \cap R| \geq |e^{(k)} \cap B|$ for $e \in E(H)$ give rise to the following inequalities

$$y_1 + y_5 \geq 0 \tag{2.16}$$
$$y_2 + y_5 \geq 0 \tag{2.17}$$
$$y_3 + y_4 \geq 0 \tag{2.18}$$
$$y_1 + y_2 + y_3 + y_4 \geq 0 \tag{2.19}$$

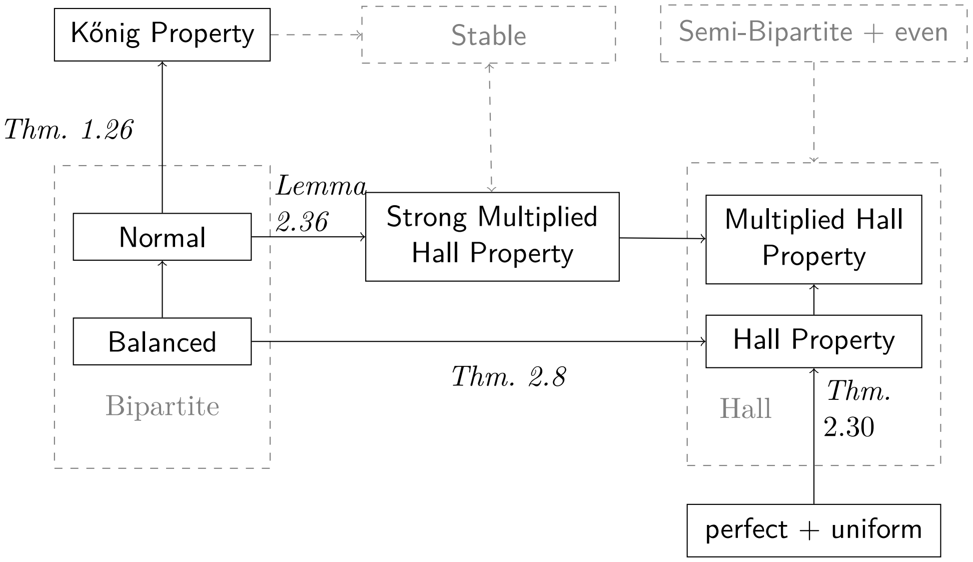


Figure 2.11: Summary of the relations between the hypergraph properties considered in this section.

The inequalities (2.15) and (2.19) imply that $y_5 < 0$. On the other hand, (2.15) together with (2.16) and (2.18) gives $y_2 < 0$. Together we get $y_2 + y_5 < 0$, contradicting (2.17). □

Table 2.11 summarizes the relationship of the hypergraph properties considered in the previous subsection; solid lines indicate hypergraph results and dashed lines an "overlay" of Table 2.5. As in the graph case, we do not draw implications implied by transitivity for better readability.

We argue that exactly the depicted implications and their transitive closure hold where we suppress trivial implications or ones that are well known. For example, all classes except "perfect + uniform" contain non-uniform hypergraphs, thus there is no arc pointing into "perfect + uniform". We go from left to right.

a) Balanced: Theorem 2.8 states that every balanced hypergraph has the Hall property. We have shown in the previous subsection that there are non-balanced hypergraphs with the Hall property, see Theorem 2.30. Furthermore, neither the multiplied nor the strong multiplied Hall property imply balancedness (take any normal, non-balanced hypergraph).

b) Normal: By Lemma 2.36, every normal hypergraph has the strong multiplied Hall property. However, there are hypergraphs with the strong multiplied Hall property that are not normal, for example, stable graphs. Every perfect uniform hypergraph that is not normal has the Hall and the multiplied Hall property. Thus, both properties do not imply normality.

c) Kőnig Property: The example in Figure 2.10 shows a hypergraph with the Kőnig property not having the multiplied Hall property. Thus, the Kőnig property does not imply the multiplied Hall property and therefore neither the Hall nor the strong multiplied Hall property. On the other hand, perfect uniform hypergraphs have the Hall and the multiplied Hall property but in general not the Kőnig property. Furthermore, every stable graph has the strong multiplied Hall property but there are stable graphs without the Kőnig property.

d) Strong Multiplied Hall Property: Clearly, the strong multiplied Hall property implies the multiplied Hall property. The reverse implication is not true. For example, the hypergraph depicted in Figure 2.7 has deficiency three but its multiplied critical difference is one. Furthermore, the strong multiplied Hall property does not imply the Hall property as every normal hypergraph has the strong multiplied Hall property but not necessarily the Hall property.

e) Multiplied Hall Property: We have already seen that there are hypergraphs with the multiplied Hall property but without the Hall property.

f) Hall Property: By Theorem 2.30 every perfect uniform hypergraph has the Hall property.

Chapter 3

Relaxed Matchings and Factors

In Chapter 2 we investigate when some specific hypergraphs have a perfect matching, i.e., when they admit a partial hypergraph in which all vertices have degree one. A way to generalize perfect matchings is to assign to every vertex a lower and an upper bound on the degree this vertex should have in a partial hypergraph. This leads to the concept of (g, f)-matchings, and the question whether or not a hypergraph has a (g, f)-matching for given functions g, f. For the case of perfect f-matchings ($g = f$) in graphs this question was answered in [Tutte, 1952]. Tutte's first proof was rather long and complicated. Later, he showed that the existence of a perfect f-matching or the existence of an f-factor can be reduced to the problem of deciding whether an auxiliary graph has a perfect matching, see [Tutte, 1954]. It is possible to generalize this approach to the (g, f)-matching case, see for example [Akiyama and Kano, 2011].

We approach the (g, f)-matching problem in hypergraphs using similar ideas. We restrict our attention to classes of hypergraphs generalizing bipartite graphs because the perfect matching problem on hypergraphs is already $\mathcal{NP}$-complete, and it is unlikely that there exists a good characterization for the existence of a perfect matching in general hypergraphs.

After a short literature overview, we look at unimodular hypergraphs in Section 3.2, where we first consider so-called relaxed f-matchings. In a relaxed f-matching we are allowed to exceed the degree bound at a vertex by paying some penalty. We define the dual concept of a relaxed b-vertex cover, where one has to pay some penalty for not covering a hyperedge often enough. We then prove a min-max theorem between relaxed f-matchings and relaxed b-vertex covers. This result is used to characterize the existence of (g, f)-matchings in unimodular hypergraphs. Section 3.2 contains joint work with Britta Peis, Oliver Schaudt, and Robert Scheidweiler. A preliminary version was published as a technical report, see [Beckenbach et al., 2017].

In Section 3.3 we investigate when a uniform hypergraph has a perfect f-matching or f-factor for Mengerian, perfect, and balanced hypergraphs. Our main results are characterizations of the existence of perfect f-matchings in uniform Mengerian hypergraphs, and f-factors in uniform balanced hypergraphs. It seems artificial to demand that all hyperedges have the same size. However, we show that it is

$\mathcal{NP}$-complete to decide whether a non-uniform balanced hypergraph has a perfect f-matching or f-factor. For non-uniform Mengerian hypergraphs it is even $\mathcal{NP}$-complete to decide whether a perfect matching exists. To the best of our knowledge complexity questions on these hypergraph classes have not been considered before. Section 3.3 is based on the article [Beckenbach and Scheidweiler, 2017], which is joint work with Robert Scheidweiler.

3.1 Literature Overview

There is a vast literature on perfect f-matchings and f-factors in graphs. We concentrate on the bipartite graph case as we generalize existence results for perfect f-matchings and f-factors in bipartite graphs to some classes of bipartite hypergraphs. For general graphs we refer to the book of Akiyama and Kano on factors, see [Akiyama and Kano, 2011], or the survey article [Plummer, 2007]. After looking at bipartite graphs, we summarize what is known about factors in hypergraphs.

The existence conditions for perfect f-matchings and f-factors given by Tutte take a much simpler form on bipartite graphs.

Theorem 3.1 (Thm. 2.4.4 [Lovász and Plummer, 1986])**.** *Given a bipartite graph G with vertex partition $V(G) = A \cup B$ and a function $f : V(G) \to \mathbb{Z}_{\geq 0}$, the graph G has a perfect f-matching if and only if*

$$f(A) = f(B) \text{ and} \tag{3.1}$$

$$f(S) \leq f(N(S)) \text{ for all } S \subseteq A, \tag{3.2}$$

where $N(S) := \{v \in V(H) \setminus S \colon v \text{ is adjacent to } s \in S\}$ denotes the neighborhood of a set of vertices.

Observe, that Theorem 3.1 is a direct generalization of Hall's theorem where the cardinality of a set is replaced by its f-value. Indeed, it can be deduced from Hall's theorem by considering the multiplication G^f of G, where each vertex v is replaced by $f(v)$ copies $v^1, \dots, v^{f(v)}$ and two vertices v^i, w^j are connected by an edge in G^f if v and w are adjacent in G. The graph G^f is still bipartite and it has a perfect matching if and only if G has a perfect f-matching. Hall's theorem applied to G^f gives rise to conditions (3.1) and (3.2) on G.

Concerning factors in bipartite graphs we get the following existence result, which is proven in [Lovász and Plummer, 1986] using the max-flow min-cut theorem.

Theorem 3.2 (Thm. 2.4.2 [Lovász and Plummer, 1986])**.** *A bipartite graph G with vertex partition $V(G) = A \cup B$ has an f-factor for a given function $f : V(G) \to \mathbb{Z}_{\geq 0}$ if and only if*

$$f(A) = f(B) \text{ and} \tag{3.3}$$

$$f(S) - f(T) \leq |\{e \in E(G) \colon e \cap S \neq \emptyset, e \cap T = \emptyset\}| \text{ for all } S \subseteq A, T \subseteq B. \tag{3.4}$$

Theorem 3.2 can be deduced from Theorem 3.1 using the reduction of the existence of an f-factor to that of a perfect f-matching described by Lovász and Plummer in Section 10.1 of [Lovász and Plummer, 1986]. Namely, each edge is subdivided into a path of length three by adding two new vertices, and f is extended to these new vertices by giving them weight one. If we start with a bipartite graph, then the resulting graph is still bipartite.

Condition (3.4) for the existence of f-factors in bipartite graphs depends on two sets in contrast to the related one for the existence of perfect f-matchings. However, it is also possible to give a one-set condition in the case of f-factors, which follows from a result of Heinrich, Hall, Kirkpatrick, and Liu in [Heinrich et al., 1990]. They characterize the existence of (g, f)-factors on bipartite graphs, and on general graphs for the special case that $g(v) < f(v)$ for all vertices v.

Theorem 3.3. *[Heinrich et al., 1990] Given a bipartite graph G, and two functions $g, f : V(G) \to \mathbb{Z}_{\geq 0}$ with $g(v) \leq f(v) \leq deg_G(v)$ for all $v \in V(G)$, the graph G has a (g, f)-factor if and only if*

$$\sum_{v \notin S} \left(g(v) - deg_{G-S}(v)\right)_+ \leq f(S) \ \forall S \subseteq V(G),$$

where $G - S$ denotes the subgraph of G induced by $V(G) \setminus S$ and $(q)_+ := \max(0, q)$ for all $q \in \mathbb{Q}$.

The decision problem whether a graph (bipartite or not) has a c-capacitated (g, f)-matching lies in $\mathcal{P}$ by a result in [Anstee, 1985]. Though Anstee's algorithm is only stated for the case $c \equiv 1$, it can also be used for the general capacities c as its running time depends only on the number of vertices and multiple edges are allowed.

In contrast to the graph case, there is only little known about factors in hypergraphs. The existing literature mainly focuses on extremal problems like Dirac-type results, see Section 2.1 of Chapter 2 and its references. Besides that, Hoffman characterizes in [Hoffman, 1960] when a system of the form

$$g \leq Ax \leq f, l \leq x \leq u \tag{3.5}$$

has a solution for a totally unimodular binary matrix A. He calls his result (which is not proven in [Hoffman, 1960]) *"the most general theorem of the Hall type"* as many Hall type theorems (existence of a system of distinct representatives, perfect matchings in bipartite graphs, flows in networks, etc.) can be reduced to it. Sixteen years later Hoffman published a proof of a slight generalization of this theorem in [Hoffman, 1976] using linear programming arguments.

Theorem 3.4. *[Hoffman, 1976] A matrix $A \in \mathbb{Q}^{m \times n}$ is totally unimodular if and only if for all $f, g \in \mathbb{Q}^m$, $l, u \in \mathbb{Q}^n$ with $g_i \leq f_i$ for $i \in [m]$ and $l_j \leq u_j$ for $j \in [n]$ the following two statements are equivalent* *(i) The system* $g \leq Ax \leq f, l \leq x \leq u$ *has a solution* $x \in \mathbb{Q}^n$.

(ii) For all $w \in \{0, \pm 1\}^m$, $v \in \{0, \pm 1\}^n$ *with* $A^T w = v$ *we have*

$$\sum_{i:w_i=-1} g_i + \sum_{j:v_j=1} l_j \leq \sum_{i:w_i=1} f_i + \sum_{j:v_j=-1} u_j.$$

Hoffman's result implies a characterization of the existence of capacitated (g, f)-matchings in unimodular hypergraphs. In terms of c-capacitated (g, f)-matchings on unimodular hypergraphs condition (ii) gives:

$$g(X) - f(Y) \leq \sum_{e \in E(H)} c(e) \left(|e \cap X| - |e \cap Y|\right)_+ \tag{3.6}$$

holds for all $X, Y \subseteq V(H)$ with $|e \cap X| - |e \cap Y| \in \{0, \pm 1\}$.

We give a combinatorial proof of this condition in Subsection 3.2.2.

3.2 Relaxed Matchings in Unimodular Hypergraphs

There are many different proofs for Kőnig's theorem in bipartite graphs. One possibility is to show that every bipartite graph has a vertex covered by every maximum size matching, and then use induction. The proof works as follows:

First, we show that in every bipartite graph with at least one edge there exists a vertex covered by every maximum matching. Suppose that G is a bipartite graph that has no such vertex, and choose any edge $e^* = \{u, v\}$ of G. Let M_u, M_v be two maximum matchings exposing u and v, respectively. We consider the multigraph G' on $V(G)$ induced by $M_u \cup M_v \cup \{e^*\}$, i.e., edges occurring both in M_u and M_v lead to parallel edges in G'. As the maximum degree of G' is two, its edge set can be partitioned into two matchings M_1, M_2 one of which must be larger than M_u or M_v. However, M_1 and M_2 form also matchings in G and therefore we get a contradiction because M_u and M_v are maximum matchings in G. Thus, there exists a vertex covered by every maximum matching.

Now, we use induction on the number of edges of a bipartite graph. A graph without edges has vertex cover number zero and matching number zero, thus the base case of the induction is trivial. If G is a bipartite graph with at least one edge, and w is a vertex covered by every maximum matching of G, then we use induction on $G - w$, which has strictly less edges than G, to obtain a vertex cover C of size $\nu(G-w)$, and observe that $C \cup \{w\}$ is a vertex cover of G of size $\nu(G-w)+1 = \nu(G)$.

Scheidweiler generalizes in his dissertation [Scheidweiler, 2011] the idea of this proof to balanced hypergraphs, where he exploits a nice coloring property of balanced hypergraphs. Together with Triesch he uses the same proof idea for a Kőnig-type theorem involving relaxed matchings in [Scheidweiler and Triesch, 2016], where a relaxed matching is a partial hypergraph in which one has to pay a penalty if a vertex is covered more than once. They use this result to give a short proof of Hall's theorem in balanced hypergraphs. Their work is the starting point of our investigation of existence criteria for (g, f)-matchings. It turns out that we have to restrict ourselves to unimodular hypergraphs, and that one not only has to relax matchings but also vertex covers. In Subsection 3.2.1 we define formally what kind of relaxation of matchings and vertex covers we are looking at, and give a min-max theorem for relaxed matchings and relaxed vertex covers in unimodular hypergraphs. Building on these results we give a combinatorial proof of Hoffman's Hall-type theorem for unimodular hypergraphs in Subsection 3.2.2.

3.2.1 Relaxed Matchings and Vertex Covers

Kőnig's theorem states that any bipartite graph admits a matching and a vertex cover of equal size. This min-max result can be extended to unimodular hypergraphs using the fact that both the matching and the vertex cover polyhedron of a unimodular hypergraph are integral. In this subsection, we give a purely combinatorial proof of a min-max theorem for unimodular hypergraphs, which can be seen as a weighted version of Kőnig's theorem, using coloring properties of unimodular hypergraphs. Our main result of this subsection, Theorem 3.13, is not new in the sense that it can also be derived from the Hoffman-Kruskal theorem (Theorem 2.2). However, the proof ideas are new and might give rise to new methods for totally unimodular matrices.

By Definition 1.18, every unimodular hypergraph admits an equitable 2-coloring, that is, a partition of its vertex set into two subsets S_1, S_2 such that every hyperedge e intersects S_1, S_2 in nearly the same number of vertices ($|e \cap S_1| - |e \cap S_2| \in \{0, 1, -1\}$). There exists also the notion of an equitable k-coloring for $k \geq 2$.

Definition 3.5 (equitable k-coloring)**.** Given an integer $k \in \mathbb{N}$ and a hypergraph H, an *equitable vertex k-coloring*, or *equitable k-coloring*, of H is a partition of $V(H)$ into k subsets $C_1, \dots, C_k$, which are called *color classes*, such that

$$\left\lfloor \frac{|e|}{k} \right\rfloor \leq |e \cap C_i| \leq \left\lceil \frac{|e|}{k} \right\rceil,$$

for all $i \in [k]$ and every hyperedge $e \in E(H)$.

It is also possible to color the hyperedges leading to the concept of an equitable edge k-coloring.

Definition 3.6 (equitable edge k-coloring)**.** Given an integer $k \in \mathbb{N}$ and a hypergraph H, an *equitable edge k-coloring* of H is a partition of $E(H)$ into k subsets

$C_1, \dots, C_k$, which are called *color classes*, such that

$$\left\lfloor \frac{\deg_H(v)}{k} \right\rfloor \leq \deg_{H[C_i]}(v) \leq \left\lceil \frac{\deg_H(v)}{k} \right\rceil,$$

for all $i \in [k]$ and every vertex $v \in V(H)$.

De Werra strengthens in [de Werra, 1971] the result on the existence of equitable 2-colorings in unimodular hypergraphs to equitable k-colorings.

Theorem 3.7. *[de Werra, 1971] A unimodular hypergraph H has an equitable vertex k-coloring for $k \geq 2$.*

The proof of this theorem is constructive and uses equitable 2-colorings to refine an arbitrary k-coloring until it is equitable.

A similar result holds for hyperedge colorings of unimodular hypergraphs, see for example Corollary 2 in Chapter 5 of [Berge, 1984]. It also follows from Theorem 3.7 by hypergraph duality. Namely, an equitable edge k-coloring in a hypergraph H corresponds to an equitable k-coloring in H^*. If H is unimodular, then also H^* is unimodular. Therefore, every unimodular hypergraph has not only an equitable k-coloring but also an equitable edge k-coloring.

Theorem 3.8. *A unimodular hypergraph H has an equitable edge k-coloring for $k \geq 2$.*

The only properties of unimodular hypergraphs that we need in the remainder of this subsection are Theorem 3.7, Theorem 3.8, and the fact that that the class of unimodular hypergraphs is closed under duality.

We consider so-called relaxed f-matchings and relaxed b-vertex covers, which are nothing but bounded functions from the hyperedge or vertex set to the non-negative integers. The key point in the definition are the associated cost functions.

Definition 3.9 (relaxed f-matching, relaxed b-vertex cover)**.** Let H be a given hypergraph together with functions $b, c : E(H) \to \mathbb{Z}_{\geq 0}$, and $f, p : V(H) \to \mathbb{Z}_{\geq 0}$.

A *c-capacitated relaxed f-matching* is a function $y : E(H) \to \mathbb{Z}_{\geq 0}$ with $y(e) \leq c(e)$ for all $e \in E(H)$. The *violation of y at v* is $\text{viol}_v(y) := (\sum_{e \in \delta_H(v)} y(e) - f(v))_+$ for every $v \in V(H)$, and the weight of y is $L(y) := b^T y - p^T \text{viol}(y)$. An *optimal* c-capacitated relaxed f-matching is one that maximizes the weight function L.

A *p-capacitated relaxed b-vertex cover* is a function $x : V(H) \to \mathbb{Z}_{\geq 0}$ with $x(v) \leq p(v)$ for all $v \in V(H)$. The *violation of x at e* is $\text{viol}_e(x) := (b(e) - \sum\limits_{v \in e} x(v))_+$ for every $e \in E(H)$, and the weight of x is $\widetilde{L}(x) := f^T x + c^T \text{viol}(x)$. A p-capacitated relaxed b-vertex cover minimizing $\widetilde{L}$ is called *optimal.*

If no confusion might occur, we call a c-capacitated relaxed f-matching just a *relaxed f-matching* and a p-capacitated relaxed b-vertex cover a *relaxed b-cover*.

Our goal in this subsection is to show that the maximum weight of a c-capacitated relaxed f-matching is equal to the minimum weight of a p-capacitated relaxed b-vertex cover on unimodular hypergraphs. Therefore, we use a similar strategy as in the alternative proof of Kőnig's Theorem described at the beginning of Section 3.2, where we argued that there exists a vertex covered by every maximum matching, and then used induction. In our case, we work on the dual side and show that there exists a hyperedge e covered at least $b(e)$-times by every optimum p-capacitated relaxed b-vertex cover, and then use induction by decreasing the capacity $c(e)$ of this hyperedge by one.

First, we prove a lemma that shows how equitable colorings can be used to improve a set of relaxed b-covers. Therefore, we need the following assumptions on the penalty costs

$$\sum_{v\in e} p(v) > b(e) \text{ for all } e \in E(H), \tag{3.7}$$

$$\sum_{e\in\delta_H(v)} c(e) > f(v) \text{ for all } v \in V(H). \tag{3.8}$$

We call the first inequality the *penalty inequality* as it states that for every hyperedge the sum of penalties for covering a vertex $v \in e$ more than $f(v)$-times is at least its b-weight $b(e)$. If this inequality does not hold, then we can always choose $y(e) = c(e)$ in an optimal relaxed f-matching as the gain $b(e)$ we get from increasing $y(e)$ is at least the penalty we have to pay for covering the vertices $v \in e$ more than $f(v)$-times.

The second inequality is called *capacity inequality* and states that for every vertex v the sum of the capacities of all hyperedges e incident to v is larger than the number of times $f(v)$ this vertex can be covered by a c-capacitated f-matching without paying any penalty. If this inequality is violated, then there exists an optimal b-vertex cover with $x(v) = 0$, as decreasing a value of $x(v) \geq 1$ by one unit changes the weight function $\widetilde{L}$ by an amount of at most $-f(v) + \sum_{e\in\delta_H(v)} c(e) \leq 0$, i.e., the weight $\widetilde{L}$ can only decrease.

The next lemma tells us how to improve a set of relaxed b-covers if no hyperedge exists such that every relaxed b-cover has violation zero at this hyperedge. In the proof of this lemma we use vertex multiplication as defined in Definition 2.33 except that we allow to multiply a vertex by zero. Multiplying a vertex v in a hypergraph H by 0 results in the subhypergraph $H \setminus v$ restricted to $V(H) \setminus \{v\}$. We show that vertex multiplication does not destroy unimodularity:

Multiplying a vertex by 0 means deleting this vertex. As every subhypergraph of a unimodular hypergraph is unimodular, deleting a vertex does not destroy unimodularity. It remains to show that a unimodular hypergraph is still unimodular after multiplying a vertex by some $\lambda \geq 1$. In the dual hypergraph, multiplying a vertex v by λ results in replacing the hyperedge e_v corresponding to v by λ parallel hyperedges. Clearly, replacing a hyperedge by some number of parallel hyperedges does not destroy the property of being unimodular. As a hypergraph is unimodular if and only if its dual is unimodular, we obtain that every vertex multiplication of a unimodular hypergraph remains unimodular.

Lemma 3.10. *Let H be a unimodular hypergraph, $b, c : E(H) \to \mathbb{Z}_{\geq 0}$, and $f, p : V(H) \to \mathbb{Z}_{\geq 0}$ be functions satisfying the capacity inequality (3.8). If there exists a vertex $v_0 \in V(H)$ contained in exactly k hyperedges $\{e_1, \ldots, e_k\} \subseteq E(H)$ and k (not necessarily distinct) p-capacitated relaxed b-covers $x^{(1)}, \ldots, x^{(k)}$ satisfying*

- *$\sum_{v \in e_i} x^{(i)}(v) < b(e_i)$ for each $i \in [k]$, and*
- *$x^{(j)}(v_0) < p(v_0)$ for at least one $j \in [k]$,*

then there is a p-capacitated relaxed b-cover x^* with $k \cdot \widetilde{L}(x^*) < \sum_{i=1}^{k} \widetilde{L}(x^{(i)})$.

Proof. Let us define a function $m : V(H) \to \mathbb{Z}_{\geq 0}$ by

$$m(v) := \begin{cases} \sum_{i=1}^{k} x^{(i)}(v) & \text{for } v \neq v_0, \\ \sum_{i=1}^{k} x^{(i)}(v_0) + 1 & \text{for } v = v_0. \end{cases}$$

We consider the hypergraph H' that arises from H by first adding singleton hyperedges $\{v\}$ for every $v \in V(H)$, and then multiplying each vertex v by $m(v)$. Both operations preserve unimodularity, thus H' is unimodular.

By Theorem 3.7, there exists an equitable k-coloring $C_1, \ldots, C_k$ of the vertices of H'. These color classes induce functions $c^{(1)}, \ldots, c^{(k)} : V(H) \to \mathbb{Z}_{\geq 0}$ by $c^{(i)}(v) = |C_i \cap \{(v, 1), \ldots, (v, m(v))\}|$ for each $v \in V(H)$ and each $i \in [k]$. That is, $c^{(i)}(v)$ is the number of copies of v contained in C_i. Each of these functions is a p-capacitated b-cover, since H' contains at most $p(v) \cdot k$ copies of each vertex $v \in V(H)$ so that the equitable k-coloring property applied on $\{(v, 1), \ldots, (v, m(v))\}$ ensures that $c^{(i)}(v) \leq \lceil m(v)/k \rceil \leq p(v)$ for all $v \in V(H)$ and $i \in [k]$.

We show that the sum of the $\widetilde{L}$-weight of the relaxed b-covers $c^{(1)}, \ldots, c^{(k)}$ is less than that of $x^{(1)}, \ldots, x^{(k)}$ implying that at least one of the p-capacitated relaxed b-covers $c^{(j)}$ must have smaller $\widetilde{L}$-weight than the average of $\widetilde{L}(x^{(i)})$ for $i \in [k]$.

By construction, we know that

$$\sum_{i=1}^{k} \sum_{v \in V(H)} f(v) \cdot c^{(i)}(v) = \sum_{i=1}^{k} \sum_{v \in V(H)} f(v) \cdot x^{(i)}(v) + f(v_0). \tag{3.9}$$

To bound the change of $\widetilde{L}$ caused by the penalties, we consider for each individual edge $g \in E(H)$ the sum of violations $\sum_{i=1}^{k} \mathrm{viol}_g(x^{(i)})$ and $\sum_{i=1}^{k} \mathrm{viol}_g(c^{(i)})$ caused at g.

We claim that none of $c^{(1)}, \dots, c^{(k)}$ has positive violation at a hyperedge $g \in E(H)$ with $\sum_{v \in g} \sum_{i=1}^{k} x^{(i)}(v) \geq k \cdot b(g)$. Using the equitable k-coloring property we get

$$\sum_{v \in g} c^{(i)}(v) = |C^i \cap g^{(m)}| \geq \left\lfloor \frac{|g^{(m)}|}{k} \right\rfloor \geq \left\lfloor \frac{\sum_{v \in g} m(v)}{k} \right\rfloor \geq b(g).$$

On the other hand, every $x^{(i)}$ has a positive violation at e_i by assumption. Thus, we get

$$\sum_{i=1}^{k} \mathrm{viol}_g(x^{(i)}) \geq \begin{cases} 1 = \sum\limits_{i=1}^{k} \mathrm{viol}_g(c^{(i)}) + 1, & \text{if } v_0 \in g \\ 0 = \sum\limits_{i=1}^{k} \mathrm{viol}_g(c^{(i)}), & \text{otherwise.} \end{cases} \tag{3.10}$$

Now, we consider hyperedges $g \in E(H)$ with $\sum_{v \in g} \sum_{i=1}^{k} x^{(i)}(v) < k \cdot b(g)$. For each such hyperedge and each $i \in [k]$ the equitable coloring property ensures that

$$\sum_{v \in g} c^{(i)}(v) \leq \left\lceil \frac{|g^{(m)}|}{k} \right\rceil \leq b(g),$$

which implies that the sum of violations at g caused by the $c^{(1)}, \dots, c^{(k)}$ is

$$\sum_{i=1}^{k} \left(b(g) - \sum_{v \in g} c^{(i)}(v) \right)_+ = \sum_{i=1}^{k} \left(b(g) - \sum_{v \in g} c^{(i)}(v) \right).$$

On the other hand, the b-covers $x^{(1)}, \dots, x^{(k)}$ cause the following violation at g

$$\sum_{i=1}^{k} \mathrm{viol}_g(x^{(i)}) = \sum_{i=1}^{k} \left(b(g) - \sum_{v \in g} x^{(i)}(v) \right)_+ \geq \sum_{i=1}^{k} \left(b(g) - \sum_{v \in g} x^{(i)}(v) \right).$$

By construction, we know that

$$\sum_{i=1}^{k} \left(b(g) - \sum_{v \in g} x^{(i)}(v) \right) = \sum_{i=1}^{k} \left(b(g) - \sum_{v \in g} c^{(i)}(v) \right) + |g \cap \{v_0\}|.$$

It follows that for every hyperedge g with $\sum_{v \in g} \sum_{i=1}^{k} x^{(i)} < k \cdot b(g)$ we have

$$\sum_{i=1}^{k} \mathrm{viol}_g(x^{(i)}) \geq \begin{cases} \sum\limits_{i=1}^{k} \mathrm{viol}_g(c^{(i)}) + 1, & \text{if } v_0 \in g \\ \sum\limits_{i=1}^{k} \mathrm{viol}_g(c^{(i)}), & \text{otherwise.} \end{cases} \tag{3.11}$$

Combining inequalities (3.9), (3.10), and (3.11) we obtain:

$$\sum_{i=1}^{k} \widetilde{L}(x^{(i)}) + f(v_0) \geq \sum_{i=1}^{k} \widetilde{L}(c_i) + \sum_{e:v_0 \in e} c(e).$$

As $\sum_{e \in \delta_H(v_0)} c(e) > f(v_0)$ by (3.8), this leads to $\sum\limits_{i=1}^{k} \widetilde{L}(x^{(i)}) > \sum\limits_{i=1}^{k} \widetilde{L}(c^{(i)})$. If we choose x^* to be the color class $c^{(j)}$ of smallest $\tilde{L}$-weight, then $k \cdot \tilde{L}(x^*) < \sum\limits_{i=1}^{k} \widetilde{L}(x^{(i)})$ as claimed. □

Lemma 3.10 implies that there exists a hyperedge e that is covered at least $b(e)$-times by every optimum b-cover if the penalty inequalities (3.7) and the capacity inequalities (3.8) hold.

Corollary 3.11. *Let H be a unimodular hypergraph, $b, c : E(H) \to \mathbb{Z}_{\geq 0}$, and $f, p : V(H) \to \mathbb{Z}_{\geq 0}$ be functions satisfying the penalty and the capacity inequalities (3.7)-(3.8). There exists a hyperedge $e \in E(H)$ such that $\sum_{v \in e} x(v) \geq b(e)$ for every optimum p-capacitated relaxed b-vertex cover.*

Proof. Suppose that no such hyperedge exists. We choose any $e_1 \in E(H)$ and an optimum p-capacitated relaxed b-vertex cover $x^{(1)}$ with $\sum_{v \in e_i} x^{(1)}(v) < b(e_1)$. As $\sum_{v \in e_1} p(v) > b(e_1)$ by (3.7), there exists a vertex $v_0 \in e_1$ with $x^{(1)}(v_0) < p(v_0)$. Let $e_2, \ldots, e_k$ be the other hyperedges of H containing v_0. For every $i = 2, \ldots, k$ let $x^{(i)}$ be an optimum p-capacitated relaxed b-vertex cover with $\sum_{v \in e_i} x^{(i)}(v) < b(e_i)$. By Lemma 3.10, there exists a p-capacitated relaxed b-cover x^* with $k \cdot \widetilde{L}(x^*) < \sum_{i=1}^{k} \widetilde{L}(x^{(i)})$; a contradiction to the optimality of $x^{(1)}, \ldots, x^{(k)}$. □

Given a hyperedge e^* that is covered at least $b(e^*)$-times be every relaxed b-vertex cover we decrease its capacity $c(e^*)$ by one. Afterwards, we have to pay less if we cover e^* less than $b(e^*)$-times. The next lemma shows that even after this change of the capacity function there exists an optimum relaxed b-vertex cover that covers e^* at least $b(e^*)$-times. This fact is needed for an inductive proof of our main theorem: Theorem 3.13.

Lemma 3.12. *Let H be a unimodular hypergraph, $b, c : E(H) \to \mathbb{Z}_{\geq 0}$, and $f, p : V(H) \to \mathbb{Z}_{\geq 0}$ be functions satisfying the capacity inequalities (3.8). Let e^* be a hyperedge with $c(e^*) \geq 1$ that is covered at least $b(e^*)$-times by every optimum relaxed b-cover. If we define a new capacity function $c' : E(H) \to \mathbb{Z}_{\geq 0}$ by $c'(e^*) := c(e^*) - 1$ and $c'(e) := c(e)$ for all other hyperedges e, then there exists an optimum relaxed b-cover $x' : V(H) \to \mathbb{Z}_{\geq 0}$ with respect to b, c', f, p such that $\sum_{v \in e^*} x'(v) \geq b(e^*)$ holds.*

Proof. Let x' be an optimum relaxed b-cover with respect to b, c', f, p covering e^* as often as possible, and let x be an optimum relaxed b-cover with respect to b, c, f, p. Suppose that $\sum_{v \in e^*} x'(v) = b(e^*) - k$ where $k \geq 1$, otherwise x' covers e^* at least $b(e^*)$-times and we are done.

We denote by $\widetilde{L}$ the weight of a relaxed b-cover with respect to the capacity function c and by $\widetilde{L}'$ the weight with respect to c'. As c and c' only differ at e^* and $\text{viol}_{e^*}(x) = 0$ we get that $\widetilde{L}'(x) = \widetilde{L}(x)$. Furthermore, we have $\widetilde{L}(x') = \widetilde{L}'(x') + k$ as $\text{viol}_{e^*}(x') = k$, and $\widetilde{L}(x) \leq \widetilde{L}(x') - 1$ because otherwise x' would be an optimal relaxed b-cover with respect to c covering e^* less than $b(e^*)$-times. In total, we get $\widetilde{L}'(x) \leq \widetilde{L}'(x') + k - 1$.

If $k = 1$, then we get $\widetilde{L}'(x) \leq \widetilde{L}'(x')$, which shows that x is also an optimal relaxed b-cover with respect to the adjusted penalty function c'. As x has violation zero at e^*, we found an optimum relaxed b-vertex cover with respect to c' covering e^* at least $b(e^*)$-times.

For $k \geq 2$ we apply a coloring trick similar as in Lemma 3.10, namely, we define a function $m(v) : V(H) \to \mathbb{Z}_{\geq 0}$ by $m(v) := (k-1)x'(v) + x(v)$ for all $v \in V(H)$, and consider the hypergraph H' that arises from H by adding singleton hyperedges $\{v\}$ for every $v \in V(H)$, and then multiplying each vertex v by $m(v)$.

Analogously as in Lemma 3.10, we choose an equitable vertex k-coloring in H', which gives us k new p-capacitated relaxed b-covers $c^{(1)}, \ldots, c^{(k)}$.

The total $\widetilde{L}'$-weight given by x and the $(k-1)$ copies of x' is

$$(k-1)\widetilde{L}'(x') + \widetilde{L}'(x) \leq (k-1)\widetilde{L}'(x') + \widetilde{L}'(x') + k - 1 = k\widetilde{L}'(x') + k - 1.$$

This shows that $\sum_{i=1}^{k} \widetilde{L}'(c^{(i)}) \leq k\widetilde{L}'(x') + k - 1$. This and the integrality of $\widetilde{L}'$ imply that there exists an index $j \in [k]$ such that $L'(c^{(j)}) \leq \widetilde{L}'(x')$. Indeed, as x' is optimal with respect to $\widetilde{L}'$ we have equality and $c^{(j)}$ is also an optimal relaxed cover. Furthermore, the inequality

$$|\,(e^*)^{(m)}\,| = \sum_{v \in e^*} m(v) \geq (k-1)(b(e^*) - k) + b(e^*) = k \cdot (b(e^*) - k + 1)$$

together with the fact that we have chosen an equitable vertex k-coloring gives

$$\sum_{v \in e^*} c^{(j)}(v) \geq \left\lfloor \frac{|(e^*)^{(m)}|}{k} \right\rfloor \geq b(e^*) - k + 1.$$

This contradicts the choice of x' and the proof is completed. □

Now, we state our main result that the cost of an optimal c-capacitated relaxed f-matching equals the cost of an optimal p-capacitated relaxed b-vertex cover.

Theorem 3.13. *Let H be a unimodular hypergraph, $b, c : E(H) \to \mathbb{Z}_{\geq 0}$, and $f, p : V(H) \to \mathbb{Z}_{\geq 0}$ be functions. The maximum weight $L(y)$ of a c-capacitated relaxed f-matching y is equal to the minimum weight $\widetilde{L}(x)$ of a p-capacitated relaxed b-cover x.*

Proof. Let y be a c-capacitated relaxed f-matching and x be a p-capacitated and relaxed b-cover. The following chain of equalities and inequalities shows that $L(y)$ is a lower bound for $\widetilde{L}(x)$.

$$\begin{aligned}
L(y) &= \sum_{e\in E(H)} y(e)b(e) - \sum_{v\in V(H)} \left(\sum_{e\in\delta_H(v)} y(e) - f(v)\right)_+ \cdot p(v) \\
&\leq \sum_{e\in E(H)} y(e)\left(\sum_{v:v\in e} x(v)\right) + \sum_{e\in E(H)} y(e)\left(b(e) - \sum_{v:v\in e} x(v)\right)_+ \\
&\quad - \sum_{v\in V(H)} \left(\sum_{e\in\delta_H(v)} y(e) - f(v)\right)_+ \cdot x(v) \\
&\leq \sum_{e\in E(H)} y(e)\left(\sum_{v:v\in e} x(v)\right) + \sum_{e\in E(H)} \left(b(e) - \sum_{v:v\in e} x(v)\right)_+ \cdot c(e) \\
&\quad - \sum_{v\in V(H)} \left(\sum_{e\in\delta_H(v)} y(e) - f(v)\right) \cdot x(v) \\
&= \sum_{v\in V(H)} x(v)f(v) + \sum_{e\in E(H)} \left(b(e) - \sum_{v:v\in e} x(v)\right)_+ \cdot c(e) \\
&= \widetilde{L}(x).
\end{aligned}$$

We prove that the maximum weight $L(y)$ of a c-capacitated relaxed f-matching y is an upper bound on the minimum weight $\widetilde{L}(x)$ of a p-capacitated relaxed b-cover x by induction over $|V(H)| + |E(H)| + \sum\limits_{e\in E(H)} c(e)$.

For the induction basis we consider the hypergraph whose vertex and hyperedge set is the empty set. Clearly, the statement of the theorem holds for this hypergraph.

For the inductive step, we assume that the theorem is true for all hypergraphs with $|V(H)| + |E(H)| + \sum_{e\in E(H)} c(e) \leq k$ and we let H be a hypergraph with $|V(H)| + |E(H)| + \sum_{e\in E(H)} c(e) = k + 1$.

If (3.7) does not hold, then there exists a hyperedge $e' \in E(H)$ with $\sum_{v:v\in e'} p(v) \leq b(e')$. In this case, we use induction on the hypergraph $H' = H[E(H) \setminus \{e'\}]$ together with the functions b, c restricted to $E(H) \setminus \{e'\}$ and f', p, where $f' : V(H') \to \mathbb{Z}_{\geq 0}$ is defined by $f'(v) := \max(0, f(v) - c(e') \cdot |\{v\} \cap e'|)$. Let x be an optimum relaxed p-cover and y be an optimum relaxed f'-matching for the instance $(H', b|_{E(H')}, c|_{E(H')}, f', p)$. We denote the weight of y by $L'(y)$ and that of x by $\widetilde{L}'(x)$. By the induction hypothesis, we know that $L'(y) = \widetilde{L}'(x)$. The function x is also a p-capacitated relaxed b-cover of H. Using that $f(v) = f'(v)$ for all $v \notin e'$ and $f(v) = f'(v) + \min(c(e'), f(v))$ for $v \in e'$, $x \leq q$, and $\sum_{v \in e'} x(v) \leq \sum_{v \in e'} p(v) \leq b(e')$ we get

$$\begin{aligned}
\widetilde{L}(x) &= \sum_{v \in V(H)} x(v)f(v) + \sum_{e \in E(H)} c(e) \left(b(e) - \sum_{v \in e} x(v) \right)_+ \\
&= \sum_{v \in V(H)} x(v)f'(v) + \sum_{v \in e'} \min\left(c(e'), f(v)\right) x(v) + \sum_{e \in E(H)} c(e) \left(b(e) - \sum_{v \in e} x(v) \right)_+ \\
&= \widetilde{L}'(x) + \sum_{v \in e'} \min\left(c(e'), f(v)\right) x(v) + c(e') \left(b(e') - \sum_{v \in e'} x(v) \right)_+ \\
&= \widetilde{L}'(x) + \sum_{v \in e'} \min\left(c(e'), f(v)\right) x(v) + c(e')b(e') - c(e') \sum_{v \in e'} x(v) \\
&\leq \widetilde{L}'(x) + c(e')b(e') + \sum_{v \in e'} \min\left(0, f(v) - c(e')\right) p(v)
\end{aligned}$$

We define a relaxed f-matching y' on H by $y'(e) = y(e)$ for all $e \in E(H')$ and $y(e') := c(e')$. The function y' is a c-capacitated relaxed f-matching of weight

$$\begin{aligned}
L(y') &= \sum_{e \in E(H)} y'(e)b(e) - \sum_{v \in V(H)} \left(\sum_{e \in \delta_H(v)} y'(e) - f(v) \right)_+ p(v) \\
&= \sum_{e \in E(H')} y(e)b(e) + c(e')b(e') - \sum_{v \in V(H) \setminus e'} \left(\sum_{e \in \delta_H(v)} y(e) - f'(v) \right)_+ p(v) \\
&\quad - \sum_{v \in e'} \left(\sum_{e \in \delta_H(v), e \neq e'} y(e) - f'(v) + c(e') - \min\left(c(e'), f(v)\right) \right)_+ p(v) \\
&\geq L'(y) + c(e')b(e') - \sum_{v \in e'} \left(c(e') - \min\left(c(e'), f(v)\right)\right) p(v) \\
&= L'(y) + c(e')b(e') + \sum_{v \in e'} \min\left(0, f(v) - c(e')\right) p(v).
\end{aligned}$$

In total, we get $L(y') \geq \widetilde{L}(x)$.

Next, we consider the case that (3.8) is violated, i.e., there exists $v \in V(H)$ with $\sum_{e \in \delta_H(v)} c(e) \leq f(v)$, then the capacity bounds $y(e) \leq c(e)$ for $e \in \delta_H(v)$ imply the degree inequality $\sum_{e \in \delta_H(v)} y(e) \leq f(v)$ for v which shows that no penalty cost occurs at v. Thus, the optimal value of a c-capacitated relaxed f-matching in H is the same as the optimal value of a c-capacitated relaxed f'-matching in $H(V(H) \setminus \{v\})$ where f' denotes the restriction of f to $V(H) \setminus \{v\}$. On the dual side, we already argued that $\sum_{e:v\in e} c(e) \leq f(v)$ implies that there exists an optimal relaxed b-cover x^* with $x^*(v) = 0$. Thus, we remove vertex v from H and apply induction on $H(V(H) \setminus \{v\})$.

In the remainder of the proof we assume that the penalty and capacity inequalities (3.7)-(3.8) hold. By Corollary 3.11, there exists a hyperedge e^* that is covered at least $b(e^*)$-times by every optimum relaxed b-cover. If $c(e^*) = 0$, then $y(e^*) = 0$ for every relaxed f-matching and for every relaxed b-cover x the penalty $c(e^*) \cdot \text{viol}_{e^*}(x)$ at e^* is zero. This shows that the optimal weight of a relaxed f-matching in H with respect to b, c, f, p is the same as the optimal weight of a relaxed f-matching in $H[E(H) \setminus \{e^*\}]$ with respect to $b|_{E(H)\setminus\{e^*\}}$, $c|_{E(H)\setminus\{e^*\}}$, f, p. The same observation holds for the optimal weight of a relaxed b-vertex cover in H and $H[E(H)\setminus\{e^*\}]$. By induction hypothesis, the optimal weight of an relaxed f-matching in $H[E(H)\setminus\{e^*\}]$ is equal to the weight of an optimal relaxed b-vertex cover.

If $c(e^*) \geq 1$, we reduce the capacity of the hyperedge e^* by one and denote the changed capacity function by c'. Furthermore, we denote the weight functions corresponding to b, c', f and p by L' and $\widetilde{L}'$. By induction hypothesis, there exist an optimum c'-capacitated relaxed f-matching y' and an optimum p-capacitated relaxed b-cover x' with $\widetilde{L}'(x') = L'(y')$. We choose x' such that it covers e^* at least $b(e^*)$-times, which is possible by Lemma 3.12. Given an optimum c-capacitated relaxed f-matching y concerning L we get

$$\begin{aligned} L(y) \geq L'(y') = \widetilde{L}'(x') &= \sum_{v\in V(H)} x'(v)f(v) + \sum_{e\in E(H)} \left(b(e) - \sum_{v:v\in e} x'(v)\right)_+ \cdot c'(e) \\ &= \sum_{v\in V(H)} x'(v)f(v) + \sum_{e\in E(H)} \left(b(e) - \sum_{v:v\in e} x'(v)\right)_+ \cdot c(e) \\ &= \widetilde{L}(x'). \end{aligned}$$

As x' is also a p-capacitated relaxed b-cover of H, this completes the proof. □

3.2.2 Existence of (g, f)-Matchings in Unimodular Hypergraphs

In this subsection we characterize the existence of c-capacitated (g, f)-matchings in unimodular hypergraphs. Therefore, we reduce the existence of a c-capacitated (g, f)-matching in a unimodular hypergraph to that of a perfect f-matching in an auxiliary unimodular hypergraph. Hence, we start by characterizing the existence of perfect f-matchings. We need the notion of vertex expansion as introduced in Definition 1.28: A hypergraph H has a perfect f-matching if and only if its expansion

H^f has a perfect matching. However, it is possible that H^f is not unimodular, even not balanced, for a unimodular hypergraph H. Thus, we cannot apply Hall's condition for balanced hypergraphs (see [Conforti et al., 1996]). However, we use Theorem 3.13 to obtain a version of Kőnig's theorem for H^f, similar to the one for perfect matchings in balanced hypergraphs obtained in [Scheidweiler and Triesch, 2016]. We apply this result to prove a direct generalization of the Hall-type condition in [Conforti et al., 1996] to perfect f-matchings in unimodular hypergraphs.

Recall, that $\nu_V(H)$ denotes the maximum number of vertices covered by a matching in H. A hypergraph has a perfect f-matching for a given function f if and only if $\nu_V(H^f) = f(V(H))$. A unimodular hypergraph is Mengerian and therefore it satisfies $\nu_V(H^f) = \tau_V(H^f)$, see Section 1.2. Thus, the non-existence of a perfect f-matching in a unimodular hypergraph H can be witnessed by a V-vertex cover $x : V(H) \to \mathbb{Z}_{\geq 0}$ with $\sum_{v \in V(H)} f(v)x(v) < f(V(H))$. In general, such a vertex cover has entries in $\{0, 1, \dots, r(H)\}$ where $r(H)$ is the rank of H, which is the maximum size of any hyperedge. In the following corollary we show that if $\nu_V(H) < f(V(H))$, then we can find a V-vertex cover of size less than $f(V(H))$ that only takes values in $\{0, 1, 2\}$. This is done by a similar approach as for perfect matchings in [Scheidweiler and Triesch, 2016].

Corollary 3.14. *Let H be a unimodular hypergraph, and $f : V(H) \to \mathbb{Z}_{\geq 0}$ be a function. We have $\nu_V(H^f) \leq f(V(H)) - k$ if and only if there exists a V-vertex cover $x : V(H) \to \mathbb{Z}_{\geq 0}$ with $x(v) \leq k+1$ for all vertices $v \in V(H)$, and $\sum_{v \in V(H)} f(v)x(v) \leq f(V(H)) - k$.*

Proof. "$\Leftarrow$": Let x be a V-vertex cover with $\sum_{v \in V(H)} f(v)x(v) \leq f(V(H)) - k$ and $y : E(H) \to \mathbb{Z}_{\geq 0}$ be an f-matching with $\sum_{e \in E(H)} |e|y(e) = \nu_V(H^f)$. Then

$$\begin{aligned} \sum_{e \in E(H)} |e|y(e) &\leq \sum_{e \in E(H)} \left(\sum_{v \in e} x(v) \right) y(e) = \sum_{v \in V(H)} x(v) \sum_{e \in \delta_H(v)} y(e) \\ &\leq \sum_{v \in V(H)} x(v)f(v) \leq f(V(H)) - k. \end{aligned}$$

"$\Rightarrow$": Apply Theorem 3.13 with $b(e) = |e|$ for all $e \in E(H)$, $p(v) = k+1$ for all $v \in V(H)$, f as given, and $c(e) = f(V(H))$ for all $e \in E(H)$. Let $y : E(H) \to \mathbb{Z}_{\geq 0}$ be an optimum relaxed f-matching. If $\mathrm{viol}_v(y) = 0$ for every $v \in V(H)$, then y is an f-matching and by assumption $\sum_{e \in E(H)} |e|f(e) \leq f(V(H)) - k$. Otherwise, we get the following bound on $L(y)$.

$$\begin{aligned}
L(y) &= \sum_{e\in E(H)} |e|y(e) - \sum_{v\in V(H)} \mathrm{viol}_v(y)\cdot(k+1)\\
&= \sum_{v\in V(H)} y(\delta_H(v)) - \sum_{v\in V(H)} (y(\delta_H(v)) - f(v))_+ \cdot (k+1)\\
&\leq \sum_{\substack{v\in V(H),\\ \mathrm{viol}_v(y)\leq 0}} f(v) + \sum_{\substack{v\in V(H),\\ \mathrm{viol}_v(y)>0}} y(\delta_H(v))\\
&\quad - \sum_{v\in V(H)} (y(\delta_H(v)) - f(v))_+ \cdot (k+1)\\
&\leq \sum_{\substack{v\in V(H),\\ \mathrm{viol}_v(y)\leq 0}} f(v) + \sum_{\substack{v\in V(H),\\ \mathrm{viol}_v(y)>0}} f(v) - k\cdot \sum_{v\in V(H)} (y(\delta_H(v)) - f(v))_+\\
&\leq f(V(H)) - k.
\end{aligned}$$

The last inequality holds because of $\mathrm{viol}_v(y) \geq 1$ for at least one $v \in V(H)$.

In both cases, Theorem 3.13 guarantees the existence of a p-capacitated relaxed b-cover $x : E(H) \to \mathbb{Z}$ of size at most $f(V(H)) - k$. As the penalty for not covering a hyperedge e at least $b(e)$-times is $f(V(H))$, x covers every hyperedge at least $b(e)$-times. Thus, x is a V-vertex cover with $x(v) \leq k + 1$ for all $v \in V(H)$ of size $\sum_{v\in V(H)} f(v)x(v) = L(y) \leq f(V(H)) - k$. □

Now, we proceed as in [Scheidweiler and Triesch, 2016].

Theorem 3.15. *Let H be a unimodular hypergraph, and $f : V(H) \to \mathbb{Z}_{\geq 0}$ be a given function. The hypergraph H has a perfect f-matching if and only if for all disjoint subsets $X, Y \subseteq V(H)$ with $f(X) > f(Y)$ there exists a hyperedge $e \in E(H)$ with $|e \cap X| > |e \cap Y|$.*

Proof. First, suppose H has a perfect f-matching y, and let $X, Y \subseteq V(H)$ be sets with $f(X) > f(Y)$, then

$$\sum_{e\in E(H)} |e\cap X|y(e) = \sum_{v\in X}\sum_{e\in\delta_H(v)} y(e) = \sum_{v\in X} f(v) > \sum_{v\in Y} f(v) = \sum_{e\in E(H)} |e\cap Y|y(e).$$

Thus, there exists a hyperedge $e \in E(H)$ with $|e \cap X| > |e \cap Y|$.

If H has no perfect f-matching, then $\nu_V(H^f) \leq f(V(H)) - 1$. By Corollary 3.14, there exists a V-vertex cover $x : V(H) \to \mathbb{Z}$ with $0 \leq x(v) \leq 2$ for all $v \in V(H)$, and $\sum_{v\in V(H)} x(v) \leq f(V(H)) - 1$. If we set $X := \{v \in V(H) \mid x(v) = 0\}$ and $Y := \{v \in V(H) \mid x(v) = 2\}$, then

$$2f(Y) + f(V(H) \setminus (X \cup Y)) < f(V(H)),$$

which implies $f(Y) < f(X)$, and for every $e \in E(H)$ we have

$$2|e \cap Y| + |e \setminus (X \cup Y)| \geq |e|,$$

which shows that $|e \cap Y| \geq |e \cap X|$ for every $e \in E(H)$. □

Finally, we reduce the capacitated (g, f)-matching problem to the perfect f-matching problem.

Corollary 3.16 (implied by [Hoffman, 1960]). *Let H be a unimodular hypergraph, and $c : E(H) \to \mathbb{Z}_{\geq 0}$, $g, f : V(H) \to \mathbb{Z}_{\geq 0}$ be functions with $g(v) \leq f(v)$ for all $v \in V(H)$.*

The hypergraph H has a c-capacitated (g, f)-matching if and only if

$$g(X) - f(Y) \leq \sum_{e \in E(H)} c(e) \left(|e \cap X| - |e \cap Y|\right)_+ \tag{3.12}$$

holds for all disjoint sets $X, Y \subseteq V(H)$.

Proof. If H has a c-capacitated (g, f)-matching y, then the following calculation shows that inequality (3.12) holds for all disjoint $X, Y \subseteq V(H)$.

$$\begin{aligned} g(X) - f(Y) &\leq \sum_{v \in X} \sum_{e \in \delta_H(v)} y(e) - \sum_{v \in Y} \sum_{e \in \delta_H(v)} y(e) \\ &= \sum_{e \in E(H)} y(e) \left(|e \cap X| - |e \cap Y|\right) \\ &\leq \sum_{e \in E(H)} y(e) \left(|e \cap X| - |e \cap Y|\right)_+ \\ &\leq \sum_{e \in E(H)} c(e) \left(|e \cap X| - |e \cap Y|\right)_+ . \end{aligned}$$

We complete the proof by showing that if H has no c-capacitated (g, f)-matching, then there exist disjoint sets $X, Y \subseteq V(H)$ that violate (3.12). Therefore, we reduce the existence of a c-capacitated (g, f)-matching in H to the existence of a perfect f-matching in an auxiliary hypergraph. For every vertex $v \in V(H)$ let v' be a copy of v, and for every hyperedge $e \in E(H)$ let v_e be a new vertex. We set $V' := \{v' : v \in V(H)\}$ and $V_E := \{v_e : e \in E(H)\}$. The auxiliary hypergraph $\widetilde{H}$ has vertex $V(\widetilde{H}) := V(H) \cup V' \cup V_E$, and hyperedges $E(\widetilde{H}) := \{e \cup \{v_e\}, \{v_e\} : e \in E(H)\} \cup \{\{v, v'\}, \{v'\} : v \in V(H)\}$. Additionally, we define a vertex function $\widetilde{f} : V(\widetilde{H}) \to \mathbb{Z}_{\geq 0}$ by

$$\widetilde{f}(\widetilde{v}) := \begin{cases} f(v) & \text{if } \widetilde{v} = v \in V(H), \\ f(v) - g(v) & \text{if } \widetilde{v} = v' \in V', \\ c(e) & \text{if } \widetilde{v} = v_e \in V_E \end{cases} .$$

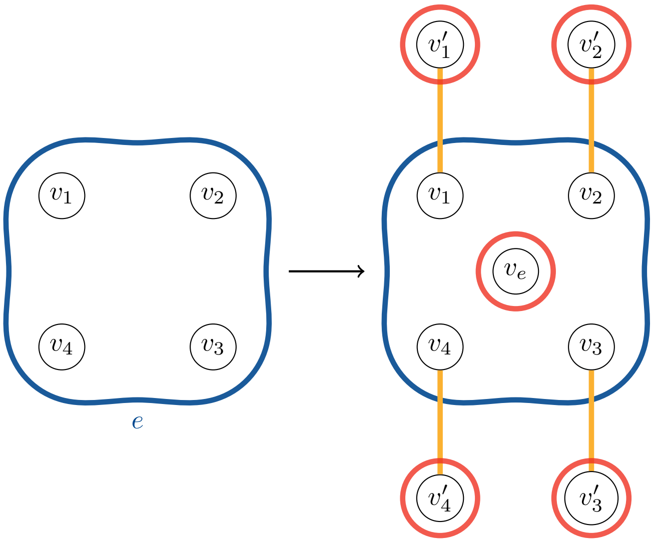


Figure 3.1: Construction of $\widetilde{H}$ from H.

Figure 3.1 illustrates the construction of $\widetilde{H}$. We show that $\widetilde{H}$ has a perfect $\widetilde{f}$-matching if and only if H has a c-capacitated (g, f)-matching.

First, let $y : E(H) \to \mathbb{Z}_{\geq 0}$ be a c-capacitated (g, f)-matching of H. We define a function $\widetilde{y}$ on $E(\widetilde{H})$ by

$$\widetilde{y}(\widetilde{e}) := \begin{cases} y\left(e\right) & \text{if } \widetilde{e} = e \cup \{v_e\}, \\ c(e) - y\left(e\right) & \text{if } \widetilde{e} = \{v_e\}, \\ f(v) - y\left(\delta_H(v)\right) & \text{if } \widetilde{e} = \{v, v'\}, \\ y\left(\delta_H(v)\right) - g(v) & \text{if } \widetilde{e} = \{v'\} \end{cases}.$$

With this definition $\widetilde{y} \geq 0$ holds as y is a c-capacitated (g, f)-matching. It remains to show that $\widetilde{y}\left(\delta_{\widetilde{H}}(\widetilde{v})\right) = \widetilde{f}(\widetilde{v})$ for all $\widetilde{v} \in V(\widetilde{H})$. If $v \in V(H)$, then

$$\widetilde{y}\left(\delta_{\widetilde{H}}(v)\right) = y\left(\delta_H(v)\right) + \widetilde{y}\left(\{v, v'\}\right) = y\left(\delta_H(v)\right) + f(v) - y\left(\delta_H(v)\right) = \widetilde{f}(v).$$

For $v' \in V'$, we have

$$\begin{aligned} \widetilde{y}\left(\delta_{\widetilde{H}}(v')\right) &= \widetilde{y}\left(\{v, v'\}\right) + \widetilde{y}\left(\{v'\}\right) \\ &= f(v) - y\left(\delta_H(v)\right) + y\left(\delta_H(v)\right) - g(v) = f(v) - g(v) \\ &= \widetilde{f}(v'), \end{aligned}$$

and for $v_e \in V_E$ we get

$$\widetilde{y}\left(\delta_{\widetilde{H}}(v_e)\right) = \widetilde{y}\left(\{v_e\}\right) + \widetilde{y}\left(e \cup \{v_e\}\right) = c(e) - y\left(e\right) + y\left(e\right) = c(e) = \widetilde{f}(v_e).$$

Thus, $\widetilde{y}$ is a perfect $\widetilde{f}$-matching of $\widetilde{H}$.

Second, let a perfect $\widetilde{f}$-matching $\widetilde{y}$ be given. If we define a function $y : E(H) \to \mathbb{Z}$ by $y(e) := \widetilde{y}\left(e \cup \{v_e\}\right)$ for all $e \in E(H)$, then y is upper bounded by c because $y(e) \leq \widetilde{y}\left(\{v_e\}\right) + \widetilde{y}\left(e \cup \{v_e\}\right) = c(e)$. Furthermore, for every $v \in V(H)$ we have

$$y\left(\delta_H(v)\right) = \widetilde{y}\left(\delta_{\widetilde{H}}(v)\right) - \widetilde{y}\left(\{v, v'\}\right) \leq \widetilde{f}(v) = f(v),$$

and

$$y\left(\delta_H(v)\right) = \widetilde{y}\left(\delta_{\widetilde{H}}(v)\right) - \widetilde{y}\left(\{v, v'\}\right) \geq f(v) - (f(v) - g(v)) = g(v),$$

where we use that $\widetilde{y}\left(\{v, v'\}\right) \leq \widetilde{y}\left(\delta_{\widetilde{H}}(v')\right) = f(v) - g(v)$. Thus, y is a c-capacitated (g, f)-matching of H.

We claim that $\widetilde{H}$ is unimodular if H is unimodular. We show that $\widetilde{H}(S)$ has an equitable 2-coloring for every $S \subseteq V(\widetilde{H})$. As H is unimodular, $H(S \cap V(H))$ has an equitable 2-coloring S_1', S_2'. We extend this coloring to an equitable 2-coloring S_1, S_2 of $\widetilde{H}(S)$. If $v_e \in S \cap V_E$ and $|e \cap S_1| \leq |e \cap S_2|$, we set $v_e \in S_1$, and otherwise $v_e \in S_2$. For every $v' \in S \cap V'$ we set $v' \in S_2$ if $v \in S_1$, and $v' \in S_1$ otherwise. The resulting sets S_1, S_2 partition S, and, by construction, they form an equitable 2-coloring of $\widetilde{H}(S)$.

If $\widetilde{H}$ has no perfect $\widetilde{f}$-matching, then there exist disjoint sets $\widetilde{X}, \widetilde{Y} \subseteq V(\widetilde{H})$ such that

$$\widetilde{f}(\widetilde{X}) > \widetilde{f}(\widetilde{Y}) \text{ and} \tag{3.13}$$

$$|\widetilde{e} \cap \widetilde{X}| \leq |\widetilde{e} \cap \widetilde{Y}| \quad \text{for all } \widetilde{e} \in E(\widetilde{H}). \tag{3.14}$$

By inequality (3.14) applied to the hyperedges of size one, $\widetilde{X}$ contains no vertices from V' and V_E, i.e., $\widetilde{X} \subseteq V(H)$. If we set $X := \widetilde{X}$ and $Y := \widetilde{Y} \cap V(H)$, then

$$\begin{aligned} |e \cap Y| &= |\left(e \cup \{v_e\}\right) \cap \widetilde{Y}| - |\{v_e\} \cap \widetilde{Y}| \geq |\left(e \cup \{v_e\}\right) \cap \widetilde{X}| - |\{v_e\} \cap \widetilde{Y}| \\ &= |e \cap X| - |\{v_e\} \cap \widetilde{Y}| \end{aligned}$$

holds for all $e \in E(H)$. This implies that $|e \cap X| - |e \cap Y| \leq 1$ with equality if and only if $v_e \in \widetilde{Y}$. Furthermore, inequality (3.14) for $\widetilde{e} = \{v, v'\}$ implies that $v' \in \widetilde{Y}$ for all $v \in X$. These observations together with (3.13) lead to

$$\begin{aligned} f(X) - f(Y) &= \widetilde{f}(\widetilde{X}) - \widetilde{f}(\widetilde{Y}) + \sum_{v' \in \widetilde{Y} \cap V'} (f(v) - g(v)) + \sum_{v_e \in \widetilde{Y} \cap V_E} c(e) \\ &> \sum_{v' \in \widetilde{Y} \cap V'} (f(v) - g(v)) + \sum_{v_e \in \widetilde{Y} \cap V_E} c(e) \\ &\geq \sum_{v \in X} (f(v) - g(v)) + \sum_{e \in E(H)} c(e) \left(|e \cap X| - |e \cap Y|\right)_+ . \end{aligned}$$

This shows that X, Y are disjoint subsets of $V(H)$ violating condition (3.12). □

3.3 Perfect f-Matchings and f-Factors

In this section we characterize the existence of perfect f-matchings in uniform hypergraphs that are Mengerian or perfect, as well as the existence of f-factors in balanced uniform hypergraphs using purely combinatorial arguments. Our results can be seen as generalizations of the conditions (3.2) and (3.4). Balanced, perfect, and Mengerian hypergraphs do not have totally unimodular incidence matrices in general. Therefore, we cannot use the results of the previous section.

The restriction to uniform hypergraphs is justified by the fact that the perfect f-matching and the f-factor problem are $\mathcal{NP}$-complete on non-uniform balanced hypergraphs, and for non-uniform Mengerian hypergraphs it is even $\mathcal{NP}$-complete to decide whether a perfect matching exists. These complexity results are proven in Section 3.4.

3.3.1 Perfect f-Matchings in Mengerian and Perfect Hypergraphs

First, we look at f-matchings in uniform Mengerian hypergraphs. Recall, that a hypergraph H is Mengerian if $\nu(H^f) = \tau(H^f)$ holds for all functions $f : V(H) \to \mathbb{Z}_{\geq 0}$. Furthermore, every Mengerian r-uniform hypergraph is r-partite by Theorem 1.30, and every balanced hypergraph is Mengerian but not the other way around.

In order to generalize Condition (3.2) stated in Section 3.1, we introduce a new concept for the neighborhood in hypergraphs. Its definition is different to the usual one in graphs. In particular, the neighborhood in a hypergraph will be a set of subsets of the vertex set and not just one subset of vertices.

Definition 3.17 (neighborhood)**.** Let H be a hypergraph and $A \subseteq V(H)$. The *neighborhood* $\mathcal{N}(A)$ of A is defined by

$$\begin{aligned}\mathcal{N}(A) := \{B \subseteq V(H) \setminus A \colon & B \text{ is (inclusionwise-)minimal such that} \\ & \text{if } e \cap A \neq \emptyset, \text{ then } e \cap B \neq \emptyset \ \forall e \in E(H)\}.\end{aligned}$$

If $e \cap A = \emptyset$ for all hyperedges $e \in E(H)$, then we define $\mathcal{N}(A) := \{\emptyset\}$.

If G is a graph and $S \subseteq V(G)$ is a stable set, then $\mathcal{N}(S)$ contains only the (graph) neighborhood $N(S)$ of S. In general hypergraphs, the neighborhood may contain more than one minimal set (compare Figure 3.2), and it is empty if there exists a hyperedge that lies completely in A.

Using the notion of a neighborhood, we state a condition that characterizes the existence of perfect f-matching. For bipartite graphs ($r = 2$) it reduces to Theorem 3.1.

Theorem 3.18. *Let H be an r-uniform Mengerian hypergraph with r-partition $V_1, \ldots, V_r$, and let $f : V(H) \to \mathbb{Z}_{\geq 0}$ be a function. The hypergraph H admits a perfect f-matching if and only if*

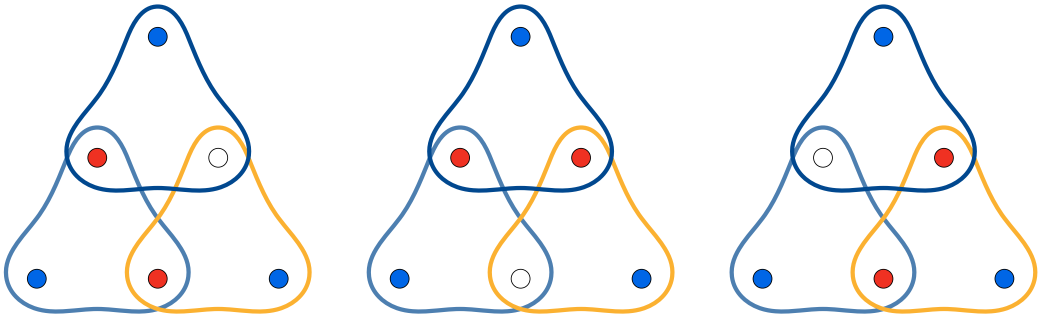


Figure 3.2: The three sets in $\mathcal{N}(S)$ (red) where S is the set of blue vertices.

(a) $f(V_1) = \ldots = f(V_r)$ *and*

(b) $f(X) \le f(Y)$ *for all* $X \subseteq V_1$, $Y \subseteq V(H) \setminus V_1$, $Y \in \mathcal{N}(X)$.

Proof. If H has a perfect f-matching $x : E(H) \to \mathbb{Z}_{\ge 0}$, then (a) holds because for every $i \in \{1, \ldots, r\}$ we have

$$f(V_i) = \sum_{v \in V_i} f(v) = \sum_{v \in V_i} \sum_{e \in \delta_H(v)} x(e) = \sum_{e \in E(H)} |e \cap V_i| x(e) = \sum_{e \in E(H)} x(e).$$

For (b), let $X \subseteq V_1$, $Y \subseteq V(H) \setminus V_1$, $Y \in \mathcal{N}(X)$. As $X \subseteq V_1$, every hyperedge intersects X in at most one vertex. Furthermore, $Y \in \mathcal{N}(X)$ implies that $|e \cap Y| \ge 1$ for every hyperedge e with $|e \cap X| = 1$. Together, we get that $|e \cap X| \le |e \cap Y|$ for all $e \in E(H)$, which yields

$$f(X) = \sum_{v \in X} \sum_{e \in \delta_H(v)} x(e) = \sum_{e \in E(H)} |e \cap X| x(e) \le \sum_{e \in E(H)} |e \cap Y| x(e) = f(Y).$$

For the other direction, suppose that H has no perfect f-matching and $f(V_i) = f^*$ for all $i \in \{1, \ldots, r\}$. Let C be an E-vertex cover of minimum f-weight. As H is Mengerian and has no perfect f-matching, we know that

$$f(C) = \tau_E(H^f) = \nu_E(H^f) < f^*.$$

Set $A_i := V_i \setminus C$ and $C^i := C \setminus V_i$ for $i = 1, \ldots, r$. We claim that there exists an index j with $f(A_j) > f(C^j)$. Otherwise,

$$f(V(H) \setminus C) = \sum_{i=1}^{r} f(A_i) \le \sum_{i=1}^{r} f(C \setminus V_i) = (r-1) f(C)$$

follows, which implies $f(C) \geq f(V(H))/r = f^*$, a contradiction to $\tau(H^f) < f^*$.

If $j \neq 1$, we have $A_1 = V_1 \setminus C = V_1 \setminus C^j$ and $C^1 = C \setminus V_1 = (C \cap V_j) \cup (C \setminus V_j) \setminus V_1 = (V_j \setminus A_j) \cup (C^j \setminus V_1)$. This together with (a) and $f(A_j) > f(C^j)$ implies that

$$\begin{aligned} f(C^1) &= f(V_j) - f(A_j) + f(C^j) - f(C^j \cap V_1) \\ &< f(V_1) - f(C^j \cap V_1) = f(A_1). \end{aligned}$$

Thus, we may assume that $j = 1$. As $e \cap A_1 \neq \emptyset$ implies $e \cap C^1 \neq \emptyset$, there exists a subset Y of C^1 with $Y \in \mathcal{N}(A_1)$. The sets $X := A_1$ and Y violate condition (b). □

For non-uniform Mengerian hypergraphs we give a necessary and a sufficient condition involving the maximum size $r(H)$ and the minimum size $s(H)$ of a hyperedge in a hypergraph H. We assume that $s(H) \geq 1$, which means that H has at least one hyperedge. The necessary condition holds even for general hypergraphs.

Lemma 3.19. *Let H be a hypergraph with $s(H) \geq 1$ and $f : V(H) \to \mathbb{Z}_{\geq 0}$ be a function on the vertices of H.*

- *(a) If H has a perfect f-matching, then $f(A) \leq (r(H)-1)f(B)$ for all $A \subseteq V(H)$ and $B \in \mathcal{N}(A)$.*
- *(b) If H is Mengerian and $f(A) \leq (s(H) - 1)f(B)$ for all sets $A \subseteq V(H)$ and $B \in \mathcal{N}(A)$, then H has a perfect f-matching.*

Proof. For (a), let H be a hypergraph with a perfect f-matching and $A \subseteq V(H)$, $B \in \mathcal{N}(A)$ be fixed sets. For every hyperedge $e \in E(H)$ intersecting A we have $|e \cap A| \leq r(H) - |e \cap B| \leq (r(H) - 1)|e \cap B|$ because $|e \cap A| \geq 1$ implies $|e \cap B| \geq 1$. If $e \in E(H)$ does not intersect A, then $|e \cap A| \leq (r(H) - 1)|e \cap B|$ holds trivially. Now, for every perfect f-matching x we have

$$f(A) = \sum_{e \in E(H)} |e \cap A| x(e) \leq (r(H) - 1) \sum_{e \in E(H)} |e \cap B| x(e) = (r - 1) f(B).$$

For (b), suppose that H is a hypergraph without a perfect f-matching and let $x : E(H) \to \mathbb{Z}_{\geq 0}$ be an f-matching maximizing $\sum_{e \in E(H)} x(e)$, in other words $\sum_{e \in E(H)} x(e) = \nu_E(H^f)$. Then $s(H) \cdot \sum_{e \in E(H)} x(e) \leq \sum_{e \in E(H)} |e| x(e) < f(V(H))$. It follows that $\nu_E(H^f) < f(V(H))/s(H)$. As H is Mengerian, there exists an E-vertex cover C of H with $f(C) < f(V(H))/s(H)$. Setting $A := V(H) \setminus C$ we get $f(A) = f(V(H)) - f(C) > (s(H) - 1)f(C)$. Because of $e \cap C \neq \emptyset$ for all $e \in E(H)$, we can choose a minimal set $B \subseteq C$ with $B \in \mathcal{N}(A)$. Clearly, also the inequality $f(A) > (s(H) - 1)f(B)$ holds. □

The constants $r(H) - 1$ and $s(H) - 1$ in (a) and (b) of the foregoing lemma are best possible as the following examples show:

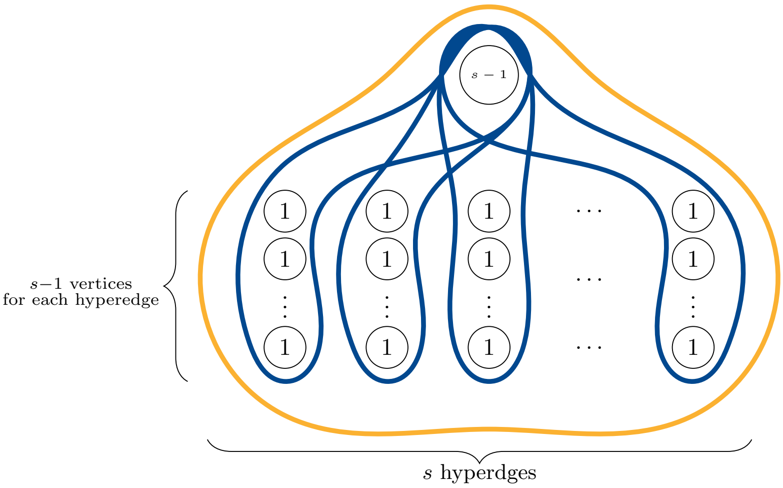


Figure 3.3: Sharpness example for Lemma 3.19 (b).

(a) Let H be the complete r-partite hypergraph on $n \cdot r$ vertices, that is, the hypergraph with $|V_1| = \ldots = |V_r| = n$, and $E(H)$ consists of all n^r hyperedges $\{v_1, \ldots, v_r\}$ with $v_1 \in V_1, \ldots, v_r \in V_r$. Clearly, H has a perfect matching. If we set $A := V(H) \setminus V_1$, then the only set in the neighborhood of A is V_1, and $|A| = (r-1)|V_1|$.

(b) For $s \geq 3$ define H_s by $V(H_s) := \{v^*\} \cup \{v_{i,j} \colon i \in [s-1], j \in [s]\}$, and $E(H_s) := \{V(H_s)\} \cup \{\{v^*, v_{1,j}, v_{2,j}, \ldots, v_{s-1,j}\} \colon j \in [s]\}$. We define a function $f : V(H_s) \to \mathbb{Z}_{\geq 0}$ by $f(v^*) := s-1$ and $f(v) := 1$ for all $v \in V(H_s) \setminus \{v^*\}$ (compare Figure 3.3). The hypergraph H_s contains no strong odd cycles. Thus, H_s is balanced and therefore also Mengerian. Furthermore, the minimum size of a hyperedge in H_s is s, and H_s admits no perfect f-matching.

We show that $f(A) \leq s \cdot f(B)$ for all $A \subseteq V(H)$ and $B \in \mathcal{N}(A)$. We distinguish whether v^* is contained in A or not. If $v^* \in A$, then B has to contain at least one vertex from $\{v_{i,j} \colon i \in [s-1]\}$ for every $j \in [s]$. This implies that $s \cdot f(B) \geq s^2 > (s-1) + s(s-2) \geq f(A)$.

If $v^* \notin A$ and $v^* \in B$, then $sf(B) \geq s(s-1) \geq f(A)$.

Finally, if $v^* \notin A$ and $v^* \notin B$, then $v_{i,j} \in A$ implies $v_{k,j} \in B$ for some $k \neq i$, and A contains at most $s-2$ vertices of $\{v_{1,j}, \ldots, v_{s-1,j}\}$ for every $j \in [s]$. It follows that $f(A) = |A| \leq (s-2)|B| = (s-2)f(B)$.

Now, we consider another class of hypergraphs generalizing bipartite graphs, namely perfect hypergraphs. Recall, that a hypergraph H is called perfect if a perfect graph G on $V(H)$ exists such that the maximal hyperedges of H correspond to the maximal cliques of G. In Chapter 2 we show that Hall's theorem for balanced hypergraph given by [Conforti et al., 1996] also holds for uniform perfect hypergraphs, see Theorem 2.30. Indeed, using a similar argument, Hall's theorem can be generalized to a characterization for the existence of perfect f-matchings by replacing the size of a set with its f-value.

Theorem 3.20. *Let H be a perfect r-uniform hypergraph and $f : V(H) \to \mathbb{Z}_{\geq 0}$ be a given function. The hypergraph H has no perfect f-matching if and only if there exists a pair $R, B \subseteq V(H)$ of disjoint vertex sets such that $|e \cap R| \geq |e \cap B|$ for every hyperedge $e \in E(H)$ holds but $f(R) < f(B)$.*

Proof. Suppose that H has a perfect f-matching x and let $R, B \subseteq V(H)$ be a pair of disjoint vertex sets such that $|e \cap R| \geq |e \cap B|$ for every $e \in E(H)$ holds. Using double-counting we obtain

$$f(R) = \sum_{v \in R} x(\delta_H(v)) = \sum_{e \in E(H)} |e \cap R| x(e) \geq \sum_{e \in E(H)} |e \cap B| x(e) = f(B).$$

Now, suppose that H has no perfect f-matching. Let G be a perfect graph on $V(H)$ such that the hyperedges of H correspond to the maximal cliques of G. As G is perfect, there exists an r-coloring of the vertices of G. If S is the color class of this coloring with the smallest f-value, then S is a stable set in G with $f(S) \leq f(V(H))/r$. On the other hand, G has a stable set $\widetilde{S}$ of f-value greater than $f(V(H))/r$. This follows from the fact that every minimum size set of cliques covering each vertex v at least $f(v)$-times must have size greater than $f(V(H))/r$, otherwise it would correspond to a perfect f-matching of H.

Now, we set $R := S \setminus \widetilde{S}$, $B := \widetilde{S} \setminus S$. It follows that

$$f(R) = f(S) - f(S \cap \widetilde{S}) < f(\widetilde{S}) - f(S \cap \widetilde{S}) = f(B).$$

By the same arguments as in the proof of Theorem 2.30, $|e \cap R| \geq |e \cap B|$ holds for every $e \in E(H)$. □

Theorem 3.20 does not hold for non-uniform perfect hypergraphs, even not in the case that $f(v) = 1$ for all vertices v as Example 2.31 in Chapter 2 shows.

3.3.2 Existence of f-Factors in Balanced Hypergraphs

In this subsection we consider the f-factor problem in balanced hypergraphs, which form a subclass of Mengerian hypergraphs. First, we give a one-set condition for the existence of f-factors in balanced uniform hypergraphs. In the case of bipartite graphs this condition is equivalent to the one stated in Theorem 3.3. Afterwards, we give a two-set condition generalizing Theorem 3.2.

In order to prove a variant of Theorem 3.3 for balanced uniform hypergraphs, we use the following min-max result given by Scheidweiler and Triesch.

Theorem 3.21. *[Scheidweiler and Triesch, 2016] Let H be a balanced hypergraph, $d : E(H) \to \mathbb{Z}_{\geq 0}$ and $b : V(H) \to \mathbb{Z}_{\geq 0}$ be functions such that $\sum_{v \in e} b(v) \geq d(e)$ holds for all $e \in E(H)$. We define the weight of a partial hypergraph H' of H by*

$$w(H') := \sum_{e \in E(H')} d(e) - \sum_{v \in V(H')} (deg_{H'}(v) - 1)_+ \, b(v).$$

Further we set

$$X := \{x \in \mathbb{Z}^{V(H)} \colon \sum_{v \in e} x(v) \geq d(e) \ \forall e \in E(H), 0 \leq x(v) \leq b(v) \ \forall v \in V(H)\}.$$

The following min-max relation holds:

$$\max_{H' \subseteq H} w(H) = \min_{x \in X} \sum_{v \in V(H)} x(v),$$

where the maximum is taken over all partial hypergraphs H' of H.

We apply this theorem to the dual hypergraph to prove the following condition for the existence of f-factors in balanced uniform hypergraphs. Recall from Theorem 1.30 that balanced r-uniform hypergraphs are r-partite.

Theorem 3.22. *If H is a balanced r-uniform hypergraph with r-partition $V_1, \ldots, V_r$, and $f : V(H) \to \mathbb{Z}_{\geq 0}$ is a function with $f(v) \leq deg_H(v)$ for all $v \in V(H)$, then H has an f-factor if and only if*

(a) $f(V_1) = \ldots = f(V_r)$ and

(b) $f(S) - f(V_1) \leq \sum_{e \in E(H)} (|e \cap S| - 1)_+$ for all $S \subseteq V(H)$.

Proof. First, suppose that H has an f-factor $F \subseteq E(H)$. For every $S \subseteq V(H)$ the following holds:

$$\begin{aligned} f(S) - f(V_1) &= \sum_{v \in S} \deg_{H[F]}(v) - \sum_{v \in V_1} \deg_{H[F]}(v) = \sum_{e \in F} |e \cap S| - \sum_{e \in F} |e \cap V_1| \\ &\leq \sum_{e \in F} (|e \cap S| - |e \cap V_1|)_+ \leq \sum_{e \in E(H)} (|e \cap S| - 1)_+ \, . \end{aligned}$$

Choosing $S = V_i$ yields $f(V_i) - f(V_1) \leq 0$. By the same argument $f(V_1) - f(V_i) \leq 0$ holds.

For the other direction, suppose that H has no f-factor and (a) holds. The optimal value of

$$\begin{aligned} \min \sum_{e \in E(H)} & x(e) \\ \sum_{e \in \delta_H(v)} x(e) &\geq f(v) \text{ for all } v \in V(H) \\ 0 \leq & x(e) \leq 1 \text{ for all } e \in E(H) \end{aligned}$$

is larger than $f(V(H))/r$ as H is r-uniform and balanced. Now, we apply Theorem 3.21 to the dual hypergraph H^* with $d = f$ and penalty costs $b(e) = 1$ for all $e \in E(H)$, which is possible as the dual of a balanced hypergraph is again balanced. By Theorem 3.21, there exists a set $S \subseteq V(H)$ with

$$f(S) - \sum_{e \in E(H)} (|e \cap S| - 1)_+ > f(V(H))/r.$$

Rearranging this inequality and using (a) yields

$$f(S) - f(V_1) > \sum_{e \in E(H)} (|e \cap S| - 1)_+ ,$$

contradicting (b). □

Now, we give an alternative characterization of the existence of f-factors in balanced uniform hypergraphs generalizing Condition (3.4) mentioned in Section 3.1.

Theorem 3.23. *If H is a balanced r-uniform hypergraph with r-partition $V_1, \dots, V_r$, and $f : V(H) \to \mathbb{Z}_{\geq 0}$ is a given function, then H has an f-factor if and only if*

(a) $f(V_1) = \ldots = f(V_r)$ and

(b) $f(X) - f(Y) \leq |\{e \in E(H) \colon e \cap X \neq \emptyset, e \cap Y = \emptyset\}|$ for all $X \subseteq V_1$ and $Y \subseteq V(H) \setminus V_1$.

Proof. If H has an f-factor, then (a) holds by Theorem 3.22. For (b) let $F \subseteq E(H)$ be an f-factor of H and $X \subseteq V_1$, $Y \subseteq V(H) \setminus V_1$ be two sets. A simple calculation gives

$$\begin{aligned} f(X) - f(Y) &= \sum_{e \in F} (|e \cap X| - |e \cap Y|) \leq \sum_{e \in F} (|e \cap X| - |e \cap Y|)_+ \\ &\leq \sum_{e \in E(H)} (|e \cap X| - |e \cap Y|)_+ \\ &= |\{e \in E(H) \colon e \cap X \neq \emptyset, e \cap Y = \emptyset\}|, \end{aligned}$$

where the last equation holds because of $|e\cap X| \leq |e\cap V_1| = 1$, and $|e\cap X|-|e\cap Y| = 1$ if and only if $e \cap X \neq \emptyset$ and $e \cap Y = \emptyset$.

For the other direction we define an auxiliary hypergraph H' with vertex set $V(H') := V(H) \cup \{v_e : e \in E(H)\}$ and hyperedges $E(H') := \{e \cup \{v_e\} : e \in E(H)\}$. Every new vertex v_e is contained in exactly one hyperedge of H' and therefore it cannot be part of a strong odd cycle. This implies that any strong odd cycle in H' is a strong odd cycle in H. As H is balanced there are no strong odd cycles in H and thus H' is balanced, too.

We define a function f' on the vertices of H' by $f'(v) := f(v)$ for all $v \in V(H)$ and $f'(v_e) := 1$ for all $v_e \in V(H') \setminus V(H)$. Every f'-matching $x' : E(H') \to \mathbb{Z}_{\geq 0}$ gives rise to an f-matching $x : E(H) \to \mathbb{Z}_{\geq 0}$ with $x \leq 1$ via $x(e) := x'(e\cup\{v_e\})$, and vice versa. Furthermore, every f'-matching x' in H' has size at most $f(V(H))/r$ because

$$\sum_{v\in V(H)} f(v) \geq \sum_{v\in V(H)} \sum_{e\in\delta_{H'}(v)} x'(e) = \sum_{e\in E(H')} |e \cap V(H)|x'(e) = r \cdot \sum_{e\in E(H')} x'(e).$$

The previous inequality implies that H' has an f'-matching x' of size $f(V(H))/r$ if and only if $x'(\delta_{H'}(v)) = f(v)$ for all $v \in V(H)$. Such an f'-matching exists if and only if H has a perfect f-matching x with $x \leq 1$, i.e., H has an f-factor.

The maximum size of an f'-matching in H' is equal to the maximum size of a matching in $H'^{f'}$. As H' is balanced and thus Mengerian, this value is the same as the minimum f'-weight of an E-vertex cover in H'. Let $C' \subseteq V(H')$ be such a vertex cover. If H has no f-factor, then $f'(C') < f(V(H))/r$. Set $C := C' \cap V(H)$, and $\widetilde{E} := \{e \in E(H)\colon v_e \in C'\}$. By the minimality of C', we know that $\widetilde{E}$ is exactly the set of hyperedges $e \in E(H)$ with $e \cap C = \emptyset$. As in the proof of Theorem 3.18, we set $A_i := V_i \setminus C$ and $C^i := C \setminus V_i$ for $i = 1, \ldots, r$. We claim that there exists an index j with

$$f(A_j) - f(C^j) > |\{e \in E(H)\colon e \cap A_j \neq \emptyset, e \cap C^j = \emptyset\}|.$$

Otherwise, we obtain

$$f(V(H) \setminus C) - (r-1)f(C) \leq \sum_{i=1}^{r} |\{e \in E(H)\colon e \cap A_i \neq \emptyset, e \cap C^i = \emptyset\}|. \tag{3.15}$$

By the definition of A_i and C^i, $e \cap A_i \neq \emptyset$ and $e \cap C^i = \emptyset$ for $e \in E(H)$ is equivalent to $e \cap C = \emptyset$. This shows that inequality (3.15) implies $f(C) + |\widetilde{E}| \geq f(V(H))/r$. However, $f(C) + |\widetilde{E}|$ is equal to $f'(C')$, which is smaller than $f(V(H))/r$. Thus, there exists and index $j \in [r]$ with

$$f(A_j) - f(C^j) > |\{e \in E(H)\colon e \cap A_j \neq \emptyset, e \cap C^j = \emptyset\}|.$$

If condition (a) holds, then by the same arguments as in the proof of Theorem 3.18 we can assume $j = 1$. In this case $X = A_1$, $Y = C^1$ violate (b). □

As multiple hyperedges are allowed, Theorem 3.23 directly implies a condition for the existence of capacitated perfect f-matchings in balanced r-uniform hypergraphs.

Corollary 3.24. *If H is a balanced r-uniform hypergraph with r-partition $V_1, \ldots, V_r$, and $f : V(H) \to \mathbb{Z}_{\geq 0}$, $c : E(H) \to \mathbb{Z}_{\geq 0}$ are functions, then H has a c-capacitated perfect f-matching if and only if*

(a) $f(V_1) = \ldots = f(V_r)$ and

(b) $f(X) - f(Y) \leq \sum_{e \in \mathcal{E}(X,Y)} c(e)$ for all $X \subseteq V_1$ and $Y \subseteq V(H) \setminus V_1$, where $\mathcal{E}(X,Y) := \{e \in E(H) : e \cap X \neq \emptyset, e \cap Y = \emptyset\}$.

Proof. Let H' be the hypergraph on the same vertex set as H, containing $c(e)$ copies of every hyperedge $e \in E(H)$. The hypergraph H has a c-capacitated perfect f-matching if and only if H has an f-factor. Thus, the stated corollary follows from Theorem 3.23. □

It is an open problem whether Theorem 3.23 and Corollary 3.24 also hold for uniform Mengerian hypergraphs. At first sight it seems like the proof of Theorem 3.23 also works for Mengerian hypergraphs as we only need that the constructed auxiliary hypergraph is Mengerian. However, the construction of the auxiliary hypergraph might destroy the Mengerian property as the following example shows.

Example 3.25. Let H be a triangle with one singleton hyperedge $e_1 = \{v_1\}$ added. The vertex-hyperedge incidence matrix of H is

$$\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}.$$

It is straightforward to see that H is Mengerian:

Given a function $f : V(H) \to \mathbb{Z}_{\geq 0}$ we show that the minimum f-weight of a vertex cover equals the maximum size of an f-matching in H. Every minimal vertex cover contains v_1 and exactly one of the vertices v_2, v_3, thus the minimum weight of a vertex cover is $f(v_1) + \min(f(v_2), f(v_3))$. On the other hand, if we take $f(v_1)$ times edge e_1 and $\min(f(v_2), f(v_3))$ times the edges $\{v_2, v_3\}$ we get an f-matching of size $f(v_1) + \min(f(v_2), f(v_3))$, i.e., $\nu_E(H^f) = \tau_E(H^f)$.

Now, we look at the hypergraph H' obtained from H by adding for every $e \in E(H)$ a new vertex v_e to the vertex set and replacing e by $e \cup \{v_e\}$. If we define a weight function f by $f(v_{e_1}) = 0$ and $f(v) = 1$ for all other vertices of H', then the minimum f-weight of an E-vertex cover is 2 while the maximum size of an f-matching is 1. So, H' is not Mengerian.

3.4 Complexity Results

In the foregoing two sections we only dealt with uniform hypergraphs, which is justified by the fact that most non-uniform existence questions turn out to be hard. Indeed, we show in the remainder of this chapter that the f-factor and perfect f-matching problem on non-uniform balanced hypergraphs, as well as the perfect matching problem on Mengerian hypergraphs are $\mathcal{NP}$-complete.

Theorem 3.26. *Let H be a balanced hypergraph and $f : V(H) \to \mathbb{Z}_{\geq 0}$ be a function on the vertices of H. Deciding whether H has an f-factor, respectively, a perfect f-matching is $\mathcal{NP}$-complete, even if $f(v) \leq 4$ for all vertices $v \in V(H)$.*

Proof. The two problems are clearly in $\mathcal{NP}$. We show that 3-dimensional matching, which is one of Karp's 21 $\mathcal{NP}$-complete problems [Karp, 1972], is reducible to the f-factor problem in balanced hypergraphs. The same reduction works for the perfect f-matching problem as well.

An instance of the 3-dimensional matching problem consists of an arbitrary 3-partite hypergraph H with vertex set $V(H) = V_1 \cup V_2 \cup V_3$ and $|V_1| = |V_2| = |V_3|$. One has to decide whether or not H admits a perfect matching.

Given an instance of the 3-dimensional matching problem we define an auxiliary balanced hypergraph $\widetilde{H}$ as follows (compare Figure 3.4).

- The vertex set $V(\widetilde{H})$ is the union of $V(H)$, $E(H)$ and four new elements $h_{e,1}$, $h_{e,2}$, $h_{e,3}$, $h_{e,4}$ for every $e \in E(H)$.
- The edge set $E(\widetilde{H})$ consists of all edges $\{v, e\}$ for all $e \in E(H)$, $v \in V(H)$ with $v \in e$, together with all edges of the form $\{h_{e,i}, e\}$ for $e \in E(H)$, $i = 1, 2, 3, 4$, and all hyperedges $\{h_{e,1}, h_{e,2}, h_{e,3}, h_{e,4}, e\}$ for $e \in E(H)$.

We define a function $f : V(\widetilde{H}) \to \mathbb{Z}_{\geq 0}$ by $f(v) = 1 = f(h_{e,i})$ for all $v \in V(H)$, $e \in E(H)$, $i = 1, 2, 3, 4$, and $f(e) := 4$ for all $e \in E(H)$.

The hypergraph $\widetilde{H}$ defined in this way is balanced because a strong odd cycle cannot contain any hyperedge of size 5, and the remaining hyperedges form a bipartite graph. It remains to show that H has an f-factor if and only if the 3-dimensional matching instance is a 'Yes' instance.

Suppose that there exists a perfect matching $M \subseteq E(H)$ of H. We define an f-factor $F \subseteq E(\widetilde{H})$ of $\widetilde{H}$ by

$$\begin{aligned} F :=& \{\{v, e\} \colon e \in M, v \in e\} \cup \{\{h_{e,1}, h_{e,2}, h_{e,3}, h_{e,4}, e\} \colon e \in M\} \\ & \cup \{\{h_{e,i}, e\} \colon e \notin M, i = 1, 2, 3, 4\}, \end{aligned}$$

i.e., for $e \in M$ take the blue edges and for $e \notin M$ the red ones in Figure 3.4. The degree of every vertex $e \in E(H)$ in $\widetilde{H}[F]$ is four and the degree of the vertices in $V(\widetilde{H}) \setminus E(H)$ is one, therefore F is an f-factor of $\widetilde{H}$.

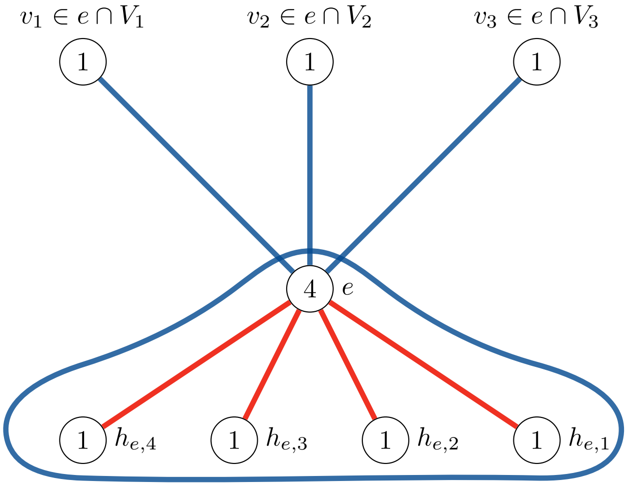


Figure 3.4: The gadget of the complexity reduction.

On the other hand, let $F \subseteq E(\widetilde{H})$ be an f-factor of $\widetilde{H}$. By the construction of $\widetilde{H}$ and f, for every $e \in E(H)$ we have either $\{h_{e,1}, h_{e,2}, h_{e,3}, h_{e,4}, e\} \in F$ or $\{h_{e,1}, e\}$, $\{h_{e,2}, e\}$, $\{h_{e,3}, e\}$, $\{h_{e,4}, e\} \in F$ but not both. We define a perfect matching M of H by

$$M := \{e \in E(H) \colon \{h_{e,1}, h_{e,2}, h_{e,3}, h_{e,4}, e\} \in F\}.$$

For every $v \in V(H)$ there exists $e \in E(H)$ such that $\{v, e\} \in F$. This implies that the edge $\{h_{e,1}, h_{e,2}, h_{e,3}, h_{e,4}, e\}$ lies in F. In particular, every $v \in V(H)$ is covered by some $e \in M$ in H. Suppose that there exist two distinct $e, e' \in M$ covering the same vertex $v \in V(H)$ in H. Looking at $\widetilde{H}$ this means that $\{h_{e,1}, h_{e,2}, h_{e,3}, h_{e,4}, e\}$, $\{h_{e',1}, h_{e',2}, h_{e',3}, h_{e',4}, e'\} \in F$, which implies $\{v, e\}$, $\{v, e'\} \in F$. However, the degree of v in $\widetilde{H}[F]$ is exactly one. Thus, M is a perfect matching of H. □

The proof of the previous theorem shows that the f-factor and the perfect f-matching problem are $\mathcal{NP}$-complete for non-uniform balanced hypergraphs of rank at least five. A balanced hypergraph of rank at most three has a totally unimodular incidence matrix, and therefore the f-factor and the perfect f-matching problem can be solved in polynomial time by linear programming. So the only open case is rank four. As there are balanced hypergraphs of rank four such that the system $Ax = f, x \geq 0$ has fractional but no integral solutions, it is likely that both problems remain $\mathcal{NP}$-complete for non-uniform balanced hypergraphs of rank four.

The perfect f-matching problem can be solved in polynomial time on uniform Mengerian hypergraphs because an r-uniform Mengerian hypergraph has a perfect f-matching if and only if $\nu_E(H^f) = f(V(H))/r$. In this case a perfect f-matching can be calculated by solving a linear program. In the non-uniform case even the perfect matching problem ($f \equiv 1$) is $\mathcal{NP}$-hard as the following corollary shows.

Corollary 3.27. *Deciding whether a Mengerian hypergraph has a perfect matching is $\mathcal{NP}$-complete.*

Proof. The problem is clearly in $\mathcal{NP}$. We show that the perfect f-matching problem in balanced hypergraphs can be reduced to it.

If H is a balanced hypergraph, and $f : V(H) \to \mathbb{Z}_{\geq 0}$ a function, then H has a perfect f-matching if and only if H^f has a perfect matching. Furthermore, H^f is Mengerian because for every function $g : V(H^f) \to \mathbb{Z}_{\geq 0}$ it holds that $(H^f)^g = H^h$, where $h : V(H) \to \mathbb{Z}_{\geq 0}$ is defined by $h(v) = \sum_{i=1,\ldots,f(v)} g(v^i)$ for all $v \in V(H)$. Thus, $\nu_E((H^f)^g) = \tau_E((H^f)^g)$ for all $g : V(H^f) \to \mathbb{Z}_{\geq 0}$. Of course, the number of vertices and edges of H^f depend on the values of f but the perfect f-matching problem in balanced hypergraphs remains $\mathcal{NP}$-complete for functions f such that $f(v) \leq 4$ for all vertices $v \in V(H)$. □

We have also seen that we can decide whether a uniform perfect hypergraph has a perfect f-matching in polynomial time. Again, we show that the perfect matching problem is $\mathcal{NP}$-hard in the non-uniform case.

Theorem 3.28. *It is $\mathcal{NP}$-complete to decide whether or not a perfect hypergraph has a perfect matching.*

Proof. The problem of deciding whether a perfect hypergraph has a perfect matching lies clearly in $\mathcal{NP}$. To show that this problem is $\mathcal{NP}$-complete, we reduce the 3-dimensional matching problem to it.

Let H be a 3-partite hypergraph. For every hyperedge e^* of H we construct a hypergraph $H(e^*)$ on the vertex set $V(H) \cup \{v_e : e \in E(H)\}$, where v_e is a new vertex representing $e \in E(H)$. The hypergraph $H(e^*)$ has a hyperedge containing all vertices of $V(H)$, for every $e \in E(H)$ is has a hyperedge $\{v_e, v_i, v_j, v_k\}$, where $e = \{v_i, v_j, v_k\}$, and for every $e \in E(H) \setminus \{e^*\}$ it has a hyperedge $\{v_e\}$ of size one. In this way, every vertex v_e for $e \in E(H) \setminus \{e^*\}$ has degree two, and v_{e^*} has degree one.

First, we observe that $H(e^*)$ is a perfect hypergraph. Therefore, let $G(H(e^*))$ be the graph obtained from $H(e^*)$ be replacing every hyperedge by a clique. The vertices of $V(H)$ form a clique in $G(H(e^*))$ and $\{v_e : e \in E(H)\}$ a stable set. Graphs with the property that their vertex set can be partitioned into a stable set and a clique are called split graph. It is known that every split graph is chordal and thus perfect, see for example [Brandstädt et al., 1999]. In particular, $G(H(e^*))$ is a perfect graph. As the maximal cliques of $G(H(e^*))$ correspond to the maximal hyperedges of $H(e^*)$, the hypergraph $H(e^*)$ is perfect.

Next, we show that $H(e^*)$ has a perfect matching if and only if H has a perfect matching containing e^*. If M is a perfect matching of $H(e^*)$, then we define a set M' by $M' := \{e \in E(H) : \{v_e\} \notin M\}$. The hyperedge e^* lies in M' because $\{v_{e^*}\} \notin E(H(e^*))$. We claim that M' is a perfect matching of H. Suppose that there exist two distinct hyperedges $e, f \in M'$ with $e \cap f \neq \emptyset$. As $e, f \in M'$, we have $\{v_e\}, \{v_f\} \notin M$. This implies that M contains the hyperedges $\{v_e\} \cup e$ and $\{v_f\} \cup f$. But then $(\{v_e\} \cup e) \cap (\{v_f\} \cup f) \neq \emptyset$, contradicting that M is a matching. Thus, M' is a matching in H. It remains to show that M' covers all vertices of H. Let v be any vertex of H, and $e \in M$ the hyperedge of M covering v, which exists as M is a perfect matching of $H(e^*)$. Thy hyperedge e cannot contain all vertices of $V(H)$, because in this case M would expose v_{e^*}. Thus e is of the form $\{v_{e'}\} \cup e'$ for some $e' \in E(H)$. This implies that $e' \in M'$ and $v \in e'$. In total, we have shown that M' is a perfect matching of H.

On the other hand, let M' be a perfect matching of H containing e^*. If we set $M := \{\{v_e\} \cup e : e \in M\} \cup \{\{v_e\} : e \notin M\}$, then M is a matching of $H(e^*)$. Furthermore, M covers every vertex $v \in V(H)$, and by construction it also covers every vertex v_e for $e \in E(H)$. Thus, M is a perfect matching of $H(e^*)$.

Now, H has a perfect matching if and only if one of the hypergraphs $H(e)$ for $e \in E(H)$ has a perfect matching. □

The perfect matching problem is a special case of the perfect f-matching, the f-factor, and the (g, f)-matching problem. Therefore, Corollary 3.27 and Theorem 3.28 imply that these three problems are $\mathcal{NP}$-complete on Mengerian and perfect hypergraphs in general. The perfect f-matching problem is polynomial time solvable if we restrict both classes to uniform hypergraphs. In contrast, we show that the (g, f)-matching problem remains $\mathcal{NP}$-complete on uniform perfect or balanced hypergraphs, and thus also on uniform Mengerian hypergraphs.

Corollary 3.29. *Let H be a uniform hypergraph that is balanced or perfect, and $f, g : V(H) \to \mathbb{Z}_{\geq 0}$ be given functions. It is $\mathcal{NP}$-complete to decide whether H has a (g, f)-matching.*

Proof. Let H be a non-uniform hypergraph and $f : V(H) \to \mathbb{Z}_{\geq 0}$ be a given function. We define an auxiliary hypergraph H' as follows. The vertex set of H' consists of $V(H)$ and new vertices $v_e^1, \dots, v_e^{r(H)-|e|}$ for every hyperedge $e \in E(H)$ of size less than the rank of H. The hypergraph H' contains all hyperedges $e \in E(H)$ of size $r(H)$ and the new hyperedges $e \cup \{v_e^1, \dots, v_e^{r(H)-|e|}\}$ for all hyperedges $e \in E(H)$ with $|e| < r(H)$. By construction, H' is uniform. Furthermore, if H is balanced, then H' is balanced, and if H is perfect, then H' is perfect.

We define functions $g', f' : V(H') \to \mathbb{Z}_{\geq 0}$ such that H' has a (g', f')-matching if and only if H has a perfect f-matching. For every $v \in V(H') \cap V(H)$ we set $g'(v) := f(v)$ and $f'(v) := f(v)$. For $v \in V(H') \setminus V(H)$ we define $g'(v) := 0$ and

$f'(v) := \infty$. With this definition it is clear that H' has a (g', f')-matching if and only if H has a perfect f-matching. □

We can decide in polynomial time whether or not an r-uniform balanced hypergraph has an f-factor by solving the linear program

$$\begin{aligned} \min \sum_{e \in E(H)} & x(e) \\ \sum_{e \in \delta_H(v)} x(e) &\geq f(v) \text{ for all } v \in V(H) \\ 0 \leq & x(e) \leq 1 \text{ for all } e \in E(H). \end{aligned}$$

Its optimal value is $f(V(H))/r$ if and only if H has an f-factor. We do not know the complexity status of the f-factor problem on uniform Mengerian or uniform perfect hypergraphs. We cannot use the linear program above as it might not be integral.

In Chapter 2 we give a Hall-type theorem for the existence of perfect matchings in normal hypergraphs. However, in this chapter we have not considered normal hypergraphs so far. As normal hypergraphs are balanced, Theorem 3.26 implies that the f-factor and the perfect f-matching problem are $\mathcal{NP}$-complete on non-uniform normal hypergraphs. In contrast to balanced hypergraphs both problems remain hard in the uniform case.

Theorem 3.30. *Let H be a uniform normal hypergraph, and $f : V(H) \to \mathbb{Z}_{\geq 0}$ be a function. Deciding whether H has an f-factor as well as deciding whether it has a perfect f-matching is $\mathcal{NP}$-complete.*

Proof. Both problems are clearly in $\mathcal{NP}$ as we can decide in polynomial time whether $x : E(H) \to \mathbb{Z}_{\geq 0}$ is a perfect f-matching or an f-factor.

We show that the perfect matching problem on an arbitrary uniform hypergraph can be reduced to the f-factor problem and the perfect f-matching problem on a uniform normal hypergraph. Given an r-uniform hypergraph H we define an $(r+1)$-uniform hypergraph H' by adding one vertex v^* and putting this vertex into all hyperedges, i.e., $V(H') := V(H) \cup \{v^*\}$ and $E(H') := \{e \cup \{v^*\} \colon e \in E(H)\}$. We observe that all hyperedges of any non-empty partial hypergraph $H'[F]$ of H' intersect in vertex v^* and therefore $H'[F]$ can be partitioned into $\Delta(H'[F])$ matchings. Thus, H' is an $(r+1)$-uniform normal hypergraph. We define a function $f : V(H) \to \mathbb{Z}_{\geq 0}$ by $f(v) = 1$ for all $v \in V(H)$ and $f(v^*) = |V(H)|/r$. With this definition H' has an f-factor if and only if H has a perfect matching. Furthermore, by the definition of f, H' has an f-factor if and only if it has a perfect f-matching. Therefore, the perfect f-matching and the f-factor problem are $\mathcal{NP}$-hard on uniform normal hypergraphs. □

We conclude this section with an overview about our complexity results for the perfect matching problem and its generalization on different classes of hypergraphs generalizing bipartite graphs in Table 3.1. We distinguish the uniform case, where all hyperedges have the same size, from the general one, where hyperedges of different sizes are allowed. $\mathcal{P}$ means that the corresponding decision problem is polynomial time solvable, and $\mathcal{NP}$ that it is $\mathcal{NP}$-complete. There are two open cases, namely the complexity of the f-factor problem on uniform hypergraphs that are Mengerian or perfect.

Hypergraph Class		Perfect Matching	Perfect f-Matching	f-Factor	(g, f)-Matching
unimodular		$\mathcal{P}$	$\mathcal{P}$	$\mathcal{P}$	$\mathcal{P}$
balanced	uniform	$\mathcal{P}$	$\mathcal{P}$	$\mathcal{P}$	$\mathcal{NP}$
	non-uniform	$\mathcal{P}$	$\mathcal{NP}$	$\mathcal{NP}$	$\mathcal{NP}$
Mengerian	uniform	$\mathcal{P}$	$\mathcal{P}$	?	$\mathcal{NP}$
	non-uniform	$\mathcal{NP}$	$\mathcal{NP}$	$\mathcal{NP}$	$\mathcal{NP}$
normal	uniform	$\mathcal{P}$	$\mathcal{NP}$	$\mathcal{NP}$	$\mathcal{NP}$
	non-uniform	$\mathcal{P}$	$\mathcal{NP}$	$\mathcal{NP}$	$\mathcal{NP}$
perfect	uniform	$\mathcal{P}$	$\mathcal{P}$	?	$\mathcal{NP}$
	non-uniform	$\mathcal{NP}$	$\mathcal{NP}$	$\mathcal{NP}$	$\mathcal{NP}$

Table 3.1: Complexity of the perfect matching problem and its generalizations in different hypergraph classes.

Chapter 4

Hypergraphs with a Perfect Matching

In Chapter 2 and 3 we give conditions for the existence of (generalized) perfect matchings on hypergraphs with some special structure generalizing bipartite graphs. In this chapter we follow a different approach. Namely, we look at the structure of connected hypergraphs in which every hyperedge is contained in a perfect matching. Graphs with this property are called matching covered. Their study emerged among others from the investigation of the perfect matching polytope. Edmonds obtained a full description of this polytope in [Edmonds, 1965a]. As it is not full-dimensional it is difficult to give a non-redundant system of linear equations and inequalities describing it. Clearly, edges not contained in any perfect matching play no role, thus only matching covered graphs are considered. Edmonds, Pulleyblank, and Lovász give in [Edmonds et al., 1982] a formula for the dimension of the perfect matching polytope of a matching covered graph in terms of the number of its vertices, its edges, and the number of so-called bricks in a tight cut decomposition. A tight cut in a matching covered graph is a cut containing exactly one edge of every perfect matching. Contracting the shores of a tight cut gives two new matching covered graphs, if they have a non-trivial tight cut, then we contract again. In this way a matching covered graph is decomposed into smaller matching covered graphs that have only trivial tight cuts where a cut is trivial if it is equal to the set of edges incident to a vertex. Such a decomposition is called a tight cut decomposition, and bricks are exactly the non-bipartite graphs without non-trivial tight cuts. Every tight cut decomposition also yields a decomposition of the perfect matching polytope, which can be used to give a minimal set of equations and inequalities describing the perfect matching polytope. This was done explicitly by Edmonds, Pulleyblank, and Lovász.

Of course, a tight cut decomposition depends on the chosen tight cuts, and it is not clear that two different decompositions yield the same list of indecomposable graphs. The result on the formula for the dimension of the perfect matching polytope implies that at least the number of bricks is always the same. Lovász proves in [Lovász, 1987] that up to multiple edges the list of resulting indecomposable graphs is always the same. We show in Section 4.4 that this remarkable result carries over to hypergraphs that can be made uniform by vertex multiplication, and give a counterexample for the general case.

We start this chapter with an overview about matching covered graphs and their tight cut decomposition in Section 4.1. Furthermore, we review known results about cuts in hypergraphs.

In Section 4.2 we look at the problem when some specific matchings can be extended to a perfect matching. First, we consider hypergraphs in which every matching of size k for some fixed k can be extended to a perfect matching. Such hypergraphs are called k-extendable. We show that a k-extendable hypergraph is also $(k-1)$-extendable if it has enough vertices, where the bound on the minimum number of vertices depends only on k and the rank of the hypergraph. Furthermore, we look at the connection between extendability and connectivity. It is known that every k-extendable graph is $(k+1)$-connected. We show that every 1-extendable uniform hypergraph is 2-connected. For $k \geq 2$ we construct k-extendable uniform hypergraphs that are not 3-connected. Afterwards, we characterize when a balanced uniform hypergraph is k-extendable. Finally, we look at hypergraphs with the property that every matching lies in a perfect matching, which we call greedily matchable. We give a polynomial time algorithm for recognizing greedily matchable hypergraphs on hypergraphs whose maximum degree is bounded by some constant.

In Section 4.3, we define tight cuts and tight cut contractions in hypergraphs and give some basic properties. To the best of our knowledge these concepts have not been considered before in hypergraphs. Our main results are that the tight cut contractions of a hypergraph induce a decomposition of its perfect matching polytope, and the uniqueness of the tight cut decomposition for hypergraphs that have a vertex multiplication that is uniform. The latter statement is proven in Section 4.4.

We conclude this chapter with some algorithmic results concerning tight cuts in Section 4.5, where we develop a polynomial time algorithm that finds a non-trivial tight cut (or decide that non exists) in a uniform balanced hypergraph.

The results of Section 4.3 and 4.4 are joint work with Meike Hatzel and Sebastian Wiederrecht.

4.1 Literature Overview

There exists a lot of literature investigating matching covered graphs and questions concerning them. We only summarize the main results and refer the interested reader to the surveys by Plummer, see [Plummer, 1994] and [Plummer, 2008].

To the best of our knowledge tight cuts have not been considered before in hypergraphs. However, there are some results on minimum cuts including a "canonical decomposition" along minimum cuts, which we summarize in the second subsection.

4.1.1 Matching Covered Graphs

A *matching covered* graph is a connected graph such that every edge is contained in a perfect matching. If k is a natural number and G is a graph with a perfect matching and at least $2k$ vertices, then G is called *k-extendable* if every matching of size k is contained in a perfect matching.

According to our definition k-extendable graphs need not be connected whereas in the literature k-extendable graphs are sometimes assumed to be connected (for example in [Plummer, 1980]).

The class of k-extendable graphs is nested with respect to increasing k. An exception occurs when k is equal to half the number of vertices of a graph because in this case k-extendability is just equivalent to the existence of a perfect matching.

Theorem 4.1. *[Plummer, 1980] If G is a connected, k-extendable graph for some positive integer $k < |V(G)|/2$, then G is $(k-1)$-extendable.*

Plummer also shows that every connected graph that is k-extendable is highly connected, except for the degenerate case that k equals half the number of vertices of a graph.

Theorem 4.2. *[Plummer, 1980] Let G be a connected graph and k be some positive integer less than $|V(G)|/2$. If G is k-extendable, then G is $(k+1)$-connected.*

In contrast to connectivity it is not easy to calculate the largest k such that a graph is k-extendable. Indeed, Koster and Hackfeld showed recently that this problem is co-$\mathcal{NP}$-hard, see [Hackfeld and Koster, 2018]. However, in bipartite graphs there exists a polynomial time algorithm to find the largest k such that the input graph is k-extendable. The first one was given by Lakhal and Litzler [Lakhal and Litzler, 1998] and the currently fastest one with a running time of $\mathcal{O}(|V(G)| \cdot |E(G)|)$ by Zhang and Zhang [Zhang and Zhang, 2006].

For fixed k there is a good characterization of k-extendability building upon Tutte's theorem on the existence of perfect matchings in graphs.

Theorem 4.3. *[Qinglin, 1993] A connected graph G is k-extendable if and only if for every $S \subseteq V(G)$ the number of odd components of $G-S$ is at most $|S|$ and the maximum size of a matching in $G[S]$ is at most i, where i is such that $G-S$ has exactly $|S|-2i$ odd components.*

For bipartite graphs there exists a simpler characterization of k-extendability generalizing Hall's theorem.

Theorem 4.4. *[McCuaig, 2001] Let G be a connected, bipartite graph with a perfect matching, and color classes A, B of the same size. If $k \in \{1, \dots, |A|-1\}$, then G is k-extendable if and only if it is connected and for every set $X \subseteq A$ with $1 \leq |X| \leq |A|-k$, we have $|N(X)| \geq |X|+k$.*

A graph G that is k-extendable for every $k = 1, \dots, |V(G)|/2$ is called *greedily matchable* or sometimes also *randomly matchable*. In other words, a graph is greedily matchable if and only if every matching lies in a perfect matching, which is the case if and only if every maximal matching is perfect. A disconnected graph is greedily matchable if and only if every connected component is greedily matchable. Thus, it suffices to characterize all connected, greedily matchable graphs. It is easy to see that the complete bipartite graph $K_{n,n}$ and the complete graph K_{2n} on an even number of vertices are greedily matchable for $n \geq 1$. Sumner proved in [Sumner, 1979] that there are no other connected, greedily matchable graphs. His result was published in 1979 before Theorem 4.1 and Theorem 4.4 were known.

Theorem 4.5. *[Sumner, 1979] A connected graph G is greedily matchable if and only if G is isomorphic to $K_{n,n}$ or K_{2n} for some $n \geq 1$.*

Though we have not found any reference about k-extendability in hypergraphs, greedily matchable hypergraphs have been considered by Caro, Sebő, and Tarsi in [Caro et al., 1996]. They give a polynomial time algorithm to decide whether a hypergraph is greedily matchable if all hyperedges have size at most r for some constant r. The algorithm uses the following characterization of greedily matchable hypergraphs.

Theorem 4.6. *[Caro et al., 1996] A hypergraph H is greedily matchable if and only if it does not contain an induced subhypergraph $H[S]$ such that*

- *$H[S]$ has a perfect matching, and*
- *there exists a hyperedge $e^* \in E(H[S])$ with $e^* \neq S$ and the property that for every perfect matching M of $H[S]$ we have $e \cap e^* \neq \emptyset$ for all $e \in M$.*

If all hyperedges have size at most r, then an induced hypergraph satisfying the second condition of the previous theorem has at most r^2 vertices. We enumerate all subhypergraphs induced by sets of size at most r^2 and test whether they satisfy the two conditions of Theorem 4.6. If r is constant this gives a polynomial time algorithm that decides whether a hypergraph of rank at most r is greedily matchable. The complexity status of the recognition problem of greedily matchable hypergraphs is open if r is part of the input. We show in Section 4.2.3 that one can also decide in polynomial time whether or not a hypergraph is greedily matchable if its maximum degree is bounded by some constant that is not part of the input.

Greedily matchable hypergraphs have also been studied in terms of their line graph. A graph G is called a *general partition graph* if it is the line graph of a greedily matchable hypergraph, and it is a *partition graph* if it is the line graph of a greedily matchable hypergraph that has no parallel hyperedges. Partition graphs were introduced by DeTemble, Harary, and Robertson in [DeTemple et al., 1987],

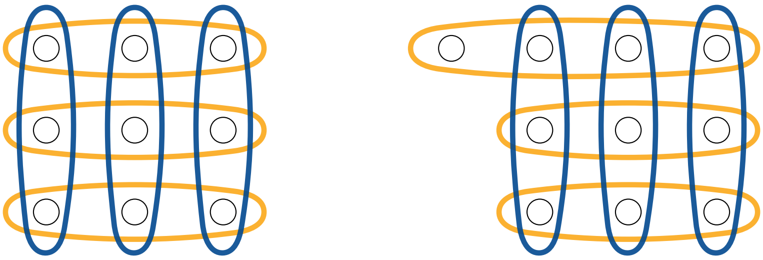


Figure 4.1: Two hypergraph with the same line graphs, where the hypergraph on the left is greedily matchable but the one on the right not.

where they also gave some necessary and sufficient conditions for a graph to be a general partition graph. More such conditions can be found in [McAvaney et al., 1993]. In particular, it is known that triangle free general partition graphs have a particular simple structure.

Theorem 4.7 ([DeTemple et al., 1987])**.** *A connected triangle free graph is a general partition graph if and only if it is isomorphic to $K_{m,n}$ for $m, n \in \mathbb{N}$.*

In particular, a bipartite general partition graph is isomorphic to a complete bipartite graphs. This fact is used to characterize 2-regular greedily matchable hypergraphs in Subsection 4.2.3.

The complexity of deciding whether a graph is a general partition graph is open. Polynomial time algorithm for the following restricted graph classes are known: chordal graphs ([Anbeek et al., 1997]), line graphs ([Levit and Milanič, 2014]), and their complements, as well as graphs of bounded maximum clique size (for the last two classes see [Hujdurović et al., 2018]). For an overview about the relation of general partition graphs to various other graph classes we refer to [Boros et al., 2017].

If a hypergraph H is greedily matchable, then its line graph $L(H)$ is a general partition graph. However, it is possible that H is not greedily matchable but $L(H)$ is a general partition graph. Namely, there might exist a hypergraph H' such that $L(H')$ and $L(H)$ are isomorphic and H' is greedily matchable but H is not. Figure 4.1 displays two hypergraphs that both have $K_{3,3}$ as their line graph. The hypergraph depicted on the left is greedily matchable, whereas the one on the right is not greedily matchable, as the blue hyperedges form a maximal matching that is not perfect. Thus, one cannot only use the line graph of a hypergraph to decide whether it is greedily matchable or not. This justifies that we look at greedily matchable hypergraphs from a hypergraphic point of view and not only at their line graphs.

Another graph class related to greedily matchable hypergraphs are so-called well-covered graphs, which were introduced in [Plummer, 1970].

Definition 4.8. A graph is *well-covered* if all maximal stable sets have the same size.

Maximal matchings in a hypergraph H correspond one-to-one to maximal independent sets in its line graph $L(H)$. Thus, $L(H)$ is well-covered if and only if all maximal matchings of H have the same size. For r-uniform hypergraphs we obtain the following relation:

Observation 4.9. *An r-uniform hypergraph H with a perfect matching is greedily-matchable if and only if $L(H)$ is well-covered.*

However, if H is not uniform, then it is possible that H is greedily matchable but $L(H)$ is not well-covered. This is the case if H has two perfect matchings of different sizes. It can also happen that $L(H)$ is well-covered and H has a perfect matching but H is not greedily matchable. Thus, in general hypergraphs we cannot reduce the characterization of greedily matchable hypergraphs to that of well-covered graphs.

Deciding whether a graph is not well-covered lies in $\mathcal{NP}$ because two maximal independent sets of different sizes are a polynomial time verifiable certificate. Indeed, the problem is $\mathcal{NP}$-complete, which was independently proven by Chvátal, and Slater in [Chvátal and Slater, 1993], and Sankaranarayana and Stewart [Sankaranarayana and Stewart, 1992], where both use a reduction from 3-SAT. Their result even holds for perfect graphs as the graphs they construct in the reduction are perfect.

Theorem 4.10. *[Chvátal and Slater, 1993][Sankaranarayana and Stewart, 1992] It is co-$\mathcal{NP}$-complete to decide whether a graph is well-covered even for perfect graphs.*

There exist characterizations of well-covered graphs in various graph classes. In particular, Dean and Zito show that there exists a polynomial time algorithm to decide whether graphs from the following classes are well-covered: trees, bipartite graphs, graphs in which the maximum size of a matching equals the minimum size of a clique cover and, additionally, no cycle of length four exists, etc. (see [Dean and Zito, 1994]). For more results on well-covered graphs we refer the interested reader to the survey [Plummer, 1993].

Now, we turn to the structure of matching covered or connected, 1-extendable graphs. Such graphs can be decomposed along tight cuts where a cut is *tight* if it intersects every perfect matching in exactly one edge. If a cut C in a graph G is given by a set of vertices S, i.e., $C = \{e \in E(G) : e = \{v, w\} \text{ with } v \in S, w \notin S\}$, then S and $V(G) \setminus S$ are called the shores of C. A non-trivial tight cut is one for which both shores have size at least two. Any tight cut yields two tight cut contractions, which are the graphs obtained by contracting one shore of the cut. We then can look for non-trivial tight cuts in the contractions and contract the shores of these cuts again. Repeating this procedure results in a list of graphs without non-trivial tight cuts, which is called a *tight cut decomposition.* A graph without non-trivial tight cuts is called a *brace* if it is bipartite, and a *brick* if it is non-bipartite. The distinction between bipartite and non-bipartite graphs without non-trivial tight cuts makes sense as bricks and braces have different properties.

It seems like it plays a huge role which cuts we choose in which order during the tight cut decomposition. This is not the case as Lovász showed.

Theorem 4.11. *[Lovász, 1987] Any two tight cut decompositions of a graph G yield the same list of bricks and braces up to parallel edges.*

The tight cut decomposition of a graph helps to analyze the perfect matching polytope, which is the convex hull of the incidence vectors of perfect matchings. Edmonds gives in [Edmonds, 1965a] a complete description of the perfect matching polytope of a graph G in terms of linear equations and inequalities, namely, it is given by

$$x_e \geq 0 \quad \forall e \in E(G) \tag{4.1}$$

$$x(\delta_G(v)) = 1 \quad \forall v \in V(G) \tag{4.2}$$

$$x(\delta_G(S)) \geq 1 \quad \forall S \subseteq V(G), |S| \geq 3, |S| \text{ is odd.} \tag{4.3}$$

The inequalities of type (4.1) are called *non-negativity constraints*, that of type (4.3) *odd-set constraints*, and the equations (4.2) are called *degree constraints.* Tight cuts correspond exactly to odd-set constraints that are satisfied with equality for every vector in the perfect matching polytope. Furthermore, every tight cut decomposition induces a decomposition of the perfect matching polytope.

Edmonds, Pulleyblank, and Lovász use in [Edmonds et al., 1982] a special tight cut decomposition, which they call brick decomposition, to determine the dimension of the perfect matching polytope and to give a minimal set of equations and inequalities describing it.

Theorem 4.12. *[Edmonds et al., 1982] Let G be a matching covered graph, and $\beta(G)$ be the number of bricks in any tight cut decomposition. The dimension of the perfect matching polytope of G is equal to $|E(G)| - |V(G)| - \beta(G) + 1$.*

Theorem 4.12 also yields a lower bound on the number of perfect matchings in a graph. Namely, as the perfect matching polytope of a graph G has dimension $|E(G)| - |V(G)| - \beta(G) + 1$, we find $|E(G)| - |V(G)| - \beta(G) + 2$ perfect matchings whose incidence vectors are linearly independent. It is hard to find the exact number of perfect matchings of a graph, even in the bipartite case, see [Valiant, 1979].

Another interesting problem is to characterize the graphs for which the perfect matching polytope is given by the non-negativity and degree constraints. It is well known that this is the case for bipartite graphs. However, there are also non-bipartite graphs with this property, for example Kőnig-Egerváry graphs, see [Kayll, 2010]. De Carvalho, Lucchesi, and Murty characterize those graphs for which the perfect matching polytope is given by the inequalities (4.1) and equations (4.2), see [de Carvalho et al., 2004]. They consider another type of cuts, which they call separating cuts. A *separating cut* in a matching covered graph G is a cut $\delta_G(S)$ such that the graphs obtained by contracting S and $V(G) \setminus S$ are matching covered. It follows that $\delta_G(S)$ is a separating cut if and only if for every edge $e \in E(G)$ there exists a perfect matching M_e with $|\delta_G(S) \cap M_e| = 1$. This implies that every tight cut is separating. The reverse implication is true for bipartite graphs but false in general. In particular, it is possible that a brick has a non-trivial separating cut, in which case the brick is called *non-solid*. A brick without a non-trivial separating cut is called a *solid brick*. De Carvalho, Lucchesi, and Murty first show that the perfect matching polytope of a brick is given by the non-negativity and degree constraints if and only if it is solid. They use this result together with the characterization of the facets of the perfect matching polytope given in [Edmonds et al., 1982] to obtain the following theorem.

Theorem 4.13. *The perfect matching polytope of a matching covered graph G is given by the non-negativity and degree constraints if and only if a tight cut decomposition of G has at most one brick and this brick is solid.*

It is not known whether the problem of deciding if a brick is solid or not can be solved in polynomial time. The only planar solid bricks are the odd wheels, where an odd wheel is an odd cycle together with one new vertex joined to all the vertices of that cycle, see [de Carvalho et al., 2006]. The characterization of non-planar solid bricks is an open problem, see [Lucchesi et al., 2018] for recent progress on this topic.

4.1.2 Cuts in Hypergraphs

In this subsection we review known results on the minimum cut problem in hypergraphs. Given a hypergraph H with non-negative weights $w \in \mathbb{Q}_{\geq 0}^{E(H)}$ a *minimum cut* is a cut $\delta_H(S)$ of H such that $\sum_{e \in \delta_H(S)} w_e$ is as small as possible. The problem of finding a minimum cut in a hypergraph can be reduced to a minimum cut computation on an auxiliary directed graph, and is thus solvable in polynomial time. However, there are also fast algorithms working directly on the hypergraph. Furthermore, there exists a decomposition along minimum cuts, which gives rise to a compact representation of all minimum cuts.

The minimum cut problem in hypergraphs occurs probably for the first time in [Lawler, 1973], where it is solved by a reduction to the minimum cut problem in directed graphs. In this article, minimum cuts in hypergraphs are used to compute an optimal partition of the vertex set into k subsets where optimality is measured in terms of different functions on the set of hyperedges intersecting at least two parts.

In [Klimmek and Wagner, 1996] a combinatorial algorithm with running time $\mathcal{O}(|V(H)|^2 \log(|V(H)|)+|V(H)|\cdot\sum_{e\in E(H)} |e|)$ for the minimum cut problem is given that works directly on the hypergraph. Furthermore, it is shown that the function $\widetilde{w} : 2^{V(H)} \to \mathbb{Q}_{\geq 0}$ defined by $\widetilde{w}(S) := w(\delta_H(S))$ for all $S \subseteq V(H)$ is submodular. In particular, all submodular function minimization algorithms can be used to compute a minimum cut in a hypergraph. Another hypergraph minimum cut algorithm with the same asymptotic running time as that of Klimmek and Wagner is given in [Mak and Wong, 2000].

There are not many other results on cuts in hypergraphs. Aissi, Mahjoub, McCormick, and Queyranne show in [Aissi et al., 2014] that the bicriteria minimum cut problem can be solved in polynomial time on graphs and hypergraphs of fixed rank. Furthermore, there are results on hypergraph k-cuts. A *k-cut* in a hypergraph is a set of hyperedges $C \subseteq E(H)$ such that $H[E(H) \setminus C]$ has at least k connected components. If k is part of the input, then it is $\mathcal{NP}$-hard to find a minimum hypergraph k-cut. Namely, Goldschmidt and Hochbaum show that the minimum k-cut problem in graphs is already $\mathcal{NP}$-hard if k is part of the input, and give a polynomial time algorithm for fixed k in [Goldschmidt and Hochbaum, 1994]. There exists a randomized polynomial time algorithm for the minimum k-cut problem in hypergraphs for constant k (see [Chandrasekaran et al., 2018]) but no deterministic polynomial time algorithm is known.

We conclude this subsection by mentioning a decomposition for hypergraphs based on contractions along minimum cuts. Cunningham and Edmonds developed a general decomposition theory for combinatorial structures, which includes for example the decomposition of graphs into 3-connected graphs. Cunningham generalizes this result in [Cunningham, 1983] to submodular functions. Namely, a submodular function f defined on subsets of a finite set E can be decomposed along so-called splits. A split is a partition $E_1 \cup E_2$ of E such that $|E_1|, |E_2| \geq 2$ and $f(E_1) + f(E_2) - f(E)$ is minimized. Both subsets of a split are contracted to obtain two new functions f_1, f_2. If f_1 or f_2 contain a split, then one continues the decomposition for f_1 and f_2 until no split exists or the functions f_1, f_2 are "highly decomposable". Cunningham uses a very restrictive notion of equivalence that is stronger than demanding the base elements to be isomorphic. He shows that there exists a unique decomposition into prime functions, which are functions without splits, and so-called brittle and semi-brittle functions, which are intuitively highly decomposable functions. Formally, a submodular function f on the subsets of E is *brittle* if every partition of E into two subsets of size at least two is a split, and it is called *semi-brittle* if there exists an ordering $e_1, \ldots, e_n$ of the elements of E such that the splits of f are exactly the partitions of the form $\{e_i, \ldots, e_{i+j-1}\}$, $\{e_{i+j}, \ldots, e_{i-1}\}$ for $i \in [n], j \in \{2, \ldots, n-2\}$, $n \geq 4$ and indices are taken modulo n.

Chekuri and Xu give in [Chekuri and Xu, 2017] a comprehensive overview of Cunningham's result applied to the minimum cut problem on hypergraphs. It gives a compact representation of all minimum cuts and can be computed in polynomial time. However, Cunningham's decomposition theory does not include our main result on the uniqueness of the tight cut decomposition stated in Theorem 4.63. It even does not imply the corresponding result for graphs by Lovász because Cunningham uses a stronger notion of equivalence than we and Lovász do. In particular, two distinct tight cut decomposition of the same graph might be non-equivalent due to Cunningham's notion.

4.2 Extendability in Hypergraphs

In this section we look at extendability problems in hypergraphs. First, we investigate k-extendable hypergraphs, where a hypergraph is k-extendable if every matching of size k can be extended to a perfect matching. In particular, we want to know when k-extendability implies $(k-1)$-extendability. Afterwards, we investigate k-extendability in balanced hypergraphs where we mainly focus on uniform hypergraphs. We give several equivalent conditions for 1-extendability generalizing known ones for 1-extendability in bipartite graphs.

In the last subsection, we consider greedily matchable hypergraphs. We show that their structure is more difficult as in the graph case, and give constructions that preserve the property of being greedily matchable. Furthermore, for every constant $d \in \mathbb{N}$ we give a polynomial time algorithm that decides whether a hypergraph of maximum degree at most d is greedily matchable.

4.2.1 k-extendable Hypergraphs

In this subsection we are looking at the question when k-extendability implies $(k-1)$-extendability, and show that a high extendability does not necessarily imply a high connectivity.

Formally, a k-extendable hypergraph is defined in the same way as a k-extendable graph.

Definition 4.14 (k-extendable hypergraph)**.** A hypergraph H is called *k-extendable* for some natural number k, if it has a matching of size k and every such matching is contained in a perfect matching.

Clearly, if H is an r-uniform hypergraph, then it can be k-extendable for k at most $|V(H)|/r$. The case $k = |V(H)|/r$ is kind of degenerate as for this value k-extendability is just equivalent to the existence of a perfect matching, thus we gain not much information in this case.

Plummer shows in [Plummer, 1980] that k-extendability in graphs implies $(k-1)$-extendability except for the degenerate case that k equals half the number of vertices. One might conjecture that a similar result also holds for r-uniform hypergraphs: If H is an r-uniform, connected, k-extendable hypergraph on more than $r \cdot k$ vertices, then H is also $(k-1)$-extendable. This is not the case as the following counterexample shows.

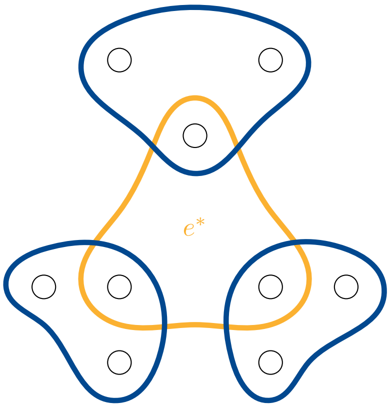


Figure 4.2: A 3-uniform, 2-extendable hypergraph that is not 1-extendable.

Example 4.15. Let H be the r-uniform hypergraph on the vertex set $\{v_1, \dots, v_{r^2}\}$ with hyperedges $e^* = \{v_1, v_{r+1}, v_{2r+1}, \dots, v_{(r-1)r+1}\}$, $e_i = \{v_{ir+1}, v_{ir+2}, \dots, v_{ir+r}\}$ for $i = 0, \dots, r-1$, see Figure 4.2 for $r = 3$. This hypergraph has r^2 vertices and is k-extendable for $k = 2, \dots, r$ because it contains exactly one perfect matching (namely, $\{e_0, \dots, e_{r-1}\}$) and every matching of size at least two is a subset of it. However, the hyperedge e^* is not contained in a perfect matching. Thus, H is not 1-extendable although it is 2-extendable and has more than $2r$ vertices.

In order to prove that k-extendability implies $(k-1)$-extendability in the graph case, Plummer shows that every matching of size $k-1$ can be extended to a matching of size k by adding one edge. Intuitively, this should also be possible for hypergraphs of bounded rank with sufficiently many vertices, and this is indeed the case.

Theorem 4.16. *If H is a k-extendable hypergraph for some $k \geq 2$ and H has more than $r(H) \cdot (r(H) + k - 2)$ vertices, then H is $(k-1)$-extendable.*

Proof. Suppose that H is k-extendable but not $(k-1)$-extendable. We show that H has at most $r(H) \cdot (r(H) + k - 2)$ vertices.

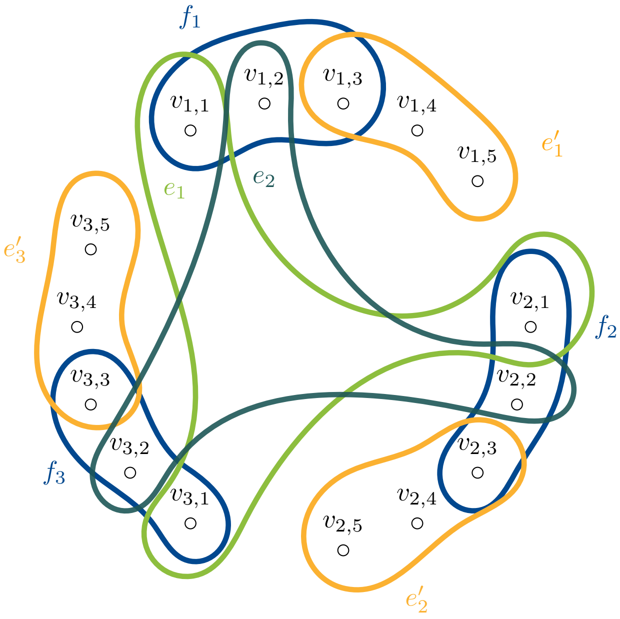


Figure 4.3: Hypergraph $H_{3,4}$ as constructed in Example 4.17.

Let $F = \{f_1, \dots, f_{k-1}\}$ be a matching of size $k-1$ that does not lie in a perfect matching of H. Choose a perfect matching M intersecting F in as many hyperedges as possible. By the choice of F we have $|M \cap F| \leq k-2$. Without loss of generality we assume that $f_1 \notin M$. Suppose that there exist hyperedges $e_1, \dots, e_t \in M \setminus F$ not intersecting f_1 with $t = k-1-|M \cap F|$. The set $\{f_1, e_1, \dots, e_t\} \cup (M \cap F)$ is a matching of size $1 + t + |M \cap F| = k$. Thus, there exists a perfect matching M' containing this set. As $|M' \cap F| = |M \cap F| + 1$, we get a contradiction to the choice of M.

We have shown that there are at most $k - 2 - |M \cap F|$ hyperedges in $M \setminus F$ disjoint from f_1. Therefore, M contains at most $|f_1| + k - 2$ hyperedges. As the size of every hyperedge of H is at most $r(H)$, we obtain

$$|V(H)| \leq r(H) \cdot |M| \leq r(H) \cdot (r(H) + k - 2) \,.$$

□

An interesting question is whether the bound on the number of vertices in Theorem 4.16 is best possible. For graphs ($r(H) = 2$) we get a bound of $2k$, which equals Plummer's result. For $k = 2$ we get $r(H)^2$, which is best possible by Example 4.15. In general, we do not know whether the bound on the number of vertices in Theorem 4.16 can be improved. The following example shows that this is not possible if $r(H) \geq k-1$. Namely, we construct a k-extendable hypergraph on $r(H) \cdot (r(H)+k-2)$ vertices that is not $(k-1)$-extendable for every $k \geq 3$ and $r(H) \geq k-1$.

Example 4.17. Given integers r, k with $k \geq 3$ and $r \geq k-1$ we define the hypergraph $H_{r,k}$ as follows. Its vertex set consists of the vertices $v_{i,j}$ for $i \in [k-1]$, $j \in [2r-1]$. If $r \geq k$ we additionally add the vertices $w_{i,j}$ for $i \in [r-1]$, $j \in [r-k+1]$ to $H_{r,k}$. The hyperedge set is the disjoint union of a perfect matching $M = \{e_1, \ldots, e_{r-1}, e'_1, \ldots, e'_{k-1}\}$ and a matching $F = \{f_1, \ldots, f_{k-1}\}$ of size $k-1$. The hyperedges of F are given by $f_i = \{v_{i,1}, \ldots, v_{i,r}\}$ for $i \in [k-1]$, and those of M by $e_i = \{v_{1,i}, \ldots, v_{k-1,i}, w_{i,1}, \ldots, w_{i,r-k+1}\}$ for $i \in [r-1]$ and $e'_i = \{v_{i,r}, \ldots, v_{i,2r-1}\}$ for $i \in [k-1]$. Figure 4.3 shows the hypergraph $H_{3,4}$.

The resulting hypergraph $H_{r,k}$ has $r \cdot |M| = r^2 + rk - 2r$ vertices and is not $(k-1)$-extendable as F is a matching of size $k-1$ that is not contained in a perfect matching. We claim that H is k-extendable. Let M' be a matching of size k. Let $f_{j_1}, \ldots, f_{j_t}$ be the hyperedges in $M' \cap F$. If $t = 0$, then $M' \subseteq M$. Otherwise, the hyperedges $e_1, \ldots, e_{r-1}$, and $e'_{j_1}, \ldots, e'_{j_t}$ have a non-empty intersection with one of the hyperedges in $M' \cap F$. It follows that

$$|M'| = |M' \cap F| + |M' \setminus F| \leq t + (r+k-2) - ((r-1)+t) = k-1,$$

contradicting $|M'| = k$. Thus, $M' \subseteq M$ and H is k-extendable.

Next, we investigate the relationship between extendability and connectivity in hypergraphs. A connected, k-extendable graph G on at least $2k+2$ vertices is $(k+1)$-connected. In particular, every 1-extendable graph on at least four vertices is 2-connected. For hypergraphs this is not always the case as the example depicted in Figure 4.4 shows. This hypergraph is balanced, connected, and 1-extendable but not 2-vertex connected.

However, we show that uniform, 1-extendable hypergraphs are 2-vertex connected. First, we consider what happens if we remove together with a vertex v all hyperedges containing v, which means that we look at $H - v$. The following lemma tells us that $\delta_H(v)$ has a special structure if $H - v$ is not connected.

Lemma 4.18. *Let H be a 1-extendable, r-uniform, connected hypergraph. If v is a vertex such that $H - v$ is not connected, and C is a connected component of $H - v$, then $\delta_H(v) = \delta_H(V(C))$.*

Proof. Let C be a connected component of $H - v$. As H is connected and $H - v$ not, there exists a hyperedge $e_C \in \delta_H(v)$ with $e_C \cap V(C) \neq \emptyset$. If M_C is a perfect matching containing e_C, then $M_C \cap E(C)$ is a maximum matching of C covering $V(C) \setminus e_C$. It follows that $|V(C)| \equiv |V(C) \cap e_C| \bmod r$ and $|V(C) \cap e_C| \geq 1$. Now,

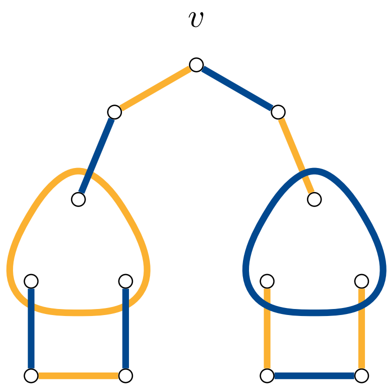


Figure 4.4: A connected, 1-extendable hypergraph and a vertex v such that $H \setminus v$ is not connected.

let $f \in \delta_H(v)$ be any other hyperedge incident to v, and M a perfect matching containing f. The set $M \cap E(C)$ is again a matching of C covering $V(C) \setminus f$. It follows that $|V(C) \cap f| = |V(C) \cap e_C| \geq 1$. This shows that $\delta_H(v) \subseteq \delta_H(V(C))$.

On the other hand, every hyperedge $e \in \delta_H(V(C))$ contains v because C is a connected component of $H - v$, thus $\delta_H(V(C)) = \delta_H(v)$. □

Using the previous lemma we prove that every 1-extendable, uniform, connected hypergraph is 2-connected. Recall, that $H \setminus v$ is the hypergraph obtained from H by deleting v and replacing all hyperedges $e \in E(H)$ by $e \setminus \{v\}$ if this set is non-empty. In the hypergraph $H - v$ we consider a "stronger" form of deletion, namely, we remove not only the vertex v from $V(H)$ but also all hyperedges e containing v from $E(H)$. In particular, if $H \setminus v$ is disconnected, then $H - v$ is also disconnected. However, it is possible that $H - v$ is not connected and $H \setminus v$ is connected.

Theorem 4.19. *Every 1-extendable, r-uniform, connected hypergraph is 2-vertex connected.*

Proof. Suppose there exists a vertex v such that $H \setminus v$ and thus $H - v$ is disconnected. Let $C_1, \dots, C_t$ be the connected components of $H - v$. All hyperedges $e \in \delta_H(v)$ intersect each C_i in at least one vertex by Lemma 4.18. Now, let $s, t \in V(H)$ be two distinct vertices. If s, t are contained in the same component C_i of $H - v$, then there exists a path connecting s to t in $H(V(H) \setminus \{v\})$. Suppose that $s \in V(C_i)$ and $t \in V(C_j)$ for $j \neq i$, and let $e \in \delta_H(v)$ arbitrary. In $H - v$ there exists a path from s to a vertex $w \in e \cap V(C_i)$ and a path from some $w' \in e \cap V(C_j)$ to t. Together with e these two paths form a path from s to t not using v. Thus, we have shown that there exists a path between every pair of distinct vertices in $H \setminus v$. Therefore, $H \setminus v$ is connected. □

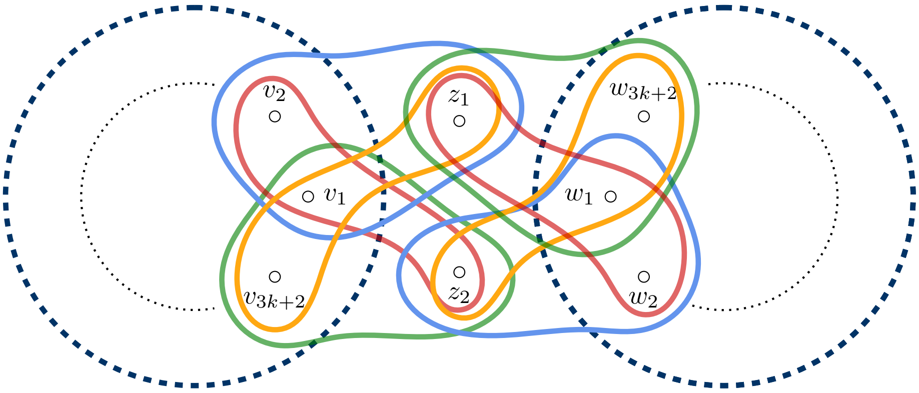


Figure 4.5: Illustration of the construction described in Example 4.20.

In view of Theorem 4.19 one might expect that at least for uniform, connected hypergraphs k-extendability implies $(k+1)$-vertex connectivity. This is not the case as the following counterexample shows.

Example 4.20. Given an integer $k \geq 1$ we consider the 3-uniform hypergraph H on the vertex set $V(H) := \{v_1, \dots, v_{3k+2}\} \cup \{w_1, \dots, w_{3k+2}\} \cup \{z_1, z_2\}$ that has the following hyperedges:

(a) All three element subsets of $\{v_2, \dots, v_{3k+2}\}$ and $\{w_2, \dots, w_{3k+2}\}$,

(b) all subsets of the form $\{z_i, v_1, v_j\}$ and $\{z_i, w_1, w_j\}$ for $j = 2, \dots, 3k + 2$ and $i = 1, 2$.

Figure 4.5 illustrates the construction of H.

The vertices $\{z_1, z_2\}$ are a separator of size two, i.e., $H \setminus \{z_1, z_2\}$ is not connected. However, every maximal matching of H is perfect. In particular, H is k'-extendable for $k' = 1, \dots, |V(H)|/3$ but not 2-vertex connected.

All in all, it seems that the structure of k-extendable hypergraphs is not as nice as the structure of k-extendable graphs. Therefore, we look at hypergraphs that are k-extendable and balanced in the following subsection.

4.2.2 Balanced Hypergraphs

In this subsection we restrict our attention to balanced uniform hypergraphs. We first consider 1-extendable hypergraphs. Afterwards, we show that the extendability of a balanced hypergraph is related to that of its dual.

Lovász and Plummer state in [Lovász and Plummer, 1986] a theorem that summarizes several known characterizations of matching covered graphs. We restate it using our notation.

Theorem 4.21 (Thm. 4.1.1 in [Lovász and Plummer, 1986])**.** *The following statements are equivalent for a connected, bipartite graph with vertex classes U, W.*

(i) G *is* 1*-extendable.*

(ii) The only minimum vertex covers of G *are* U *and* W*.*

(iii) $|U| = |W|$ *and* $|N(X)| \geq |X| + 1$ *for every non-empty set* $X \subsetneq U$*.*

(iv) $G = K_2$*, or* $|V(G)| \geq 4$*, and for any* $u \in U$*,* $w \in W$ *the subgraph* $G - u - w$ *has a perfect matching.*

(v) If $F \subseteq E(G)$ *is the set of edges that are contained in some perfect matching of* G*, then* $G[F]$ *is a connected subgraph of* G*.*

We generalize the first four statements of this theorem to ones about hypergraphs, and show that they are equivalent on uniform, balanced hypergraphs, where we use the hypergraphic neighborhood defined in Definition 3.17 and the fact that every r-uniform, balanced hypergraph is r-partite by Theorem 1.30.

Theorem 4.22. *Let* H *be a connected, balanced,* r*-uniform hypergraph with* r*-partition* $V_1, \dots, V_r$*. The following statements are equivalent.*

(a) H *is* 1*-extendable.*

(b) Every minimum E*-vertex cover is a maximum stable set.*

(c) Every maximum stable set is a minimum E*-vertex cover.*

(d) *(1)* $|V_i| = |V_j|$ *for all* $i, j \in \{1, \dots, r\}$*,*

(2) $|X| \leq |Y|$ *for all* $X \subseteq V_1$*,* $Y \subseteq V(H) \setminus V_1$*,* $Y \in \mathcal{N}(X)$ *with* $(V_1 \setminus X) \cup Y$ *a stable set, and*

(3) $|X| + 1 \leq |Y|$ *for all* $X \subseteq V_1$*,* $Y \subseteq V(H) \setminus V_1$*,* $Y \in \mathcal{N}(X)$ *with* $(V_1 \setminus X) \cup Y$ *not a stable set.*

(e) Either H *consists of* r *vertices and all hyperedges are of the form* $e = V(H)$*, or* $H - F$ *has a perfect matching for every* $F \subseteq V(H)$ *of size* r *with* $|F \cap T| = 1$ *for all stable,* E*-vertex covers* T *of* H*.*

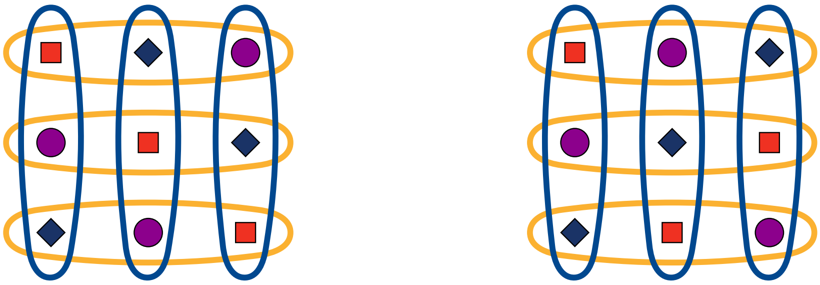


Figure 4.6: A 3-partite, balanced hypergraph with two distinct 3-partition where the vertex classes V_1, V_2, V_3 are indicated by different shapes and colors of the vertices.

The conditions (b) and (c) of Theorem 4.22 are a hypergraphic generalization of condition (ii) of Theorem 4.21 because in a connected bipartite graph G its color classes are the only minimum E-vertex covers that are also maximum stable sets. The statement (d) applied on a connected bipartite graph gives (iii) of Theorem 4.21 as in this case $(V_1 \setminus X) \cup Y$ is stable if and only if $X = V_1$ or $X = \emptyset$.

When generalizing (iv) one has to be careful. In contrast to bipartite graphs, an r-uniform, connected, balanced hypergraph might have distinct r-partitions of its vertex set, see for example Figure 4.6. Thus, we are only allowed to remove sets of vertices of size r that intersect every possible vertex class in exactly one vertex. Every vertex class is a stable E-vertex cover. On the other hand, if T is a stable E-vertex cover in an r-partite hypergraph H, then T together with an $(r-1)$-partition of $H(V(H) \setminus T)$ is an r-partition of H. Thus, we can equivalently look at sets that intersect every stable E-vertex cover in exactly one vertex. This gives an intuition why (e) is a suitable hypergraphic generalization for (iv). On the other hand, (e) applied to a connected bipartite graph with vertex classes U and W gives condition (iv) of Theorem 4.21 as U and W are the only stable, E-vertex covers, and thus every set F as in (e) contains exactly one vertex from U and one from W.

Regarding condition (v), we do not know whether a connected, uniform, balanced hypergraph H is 1-extendable, if the partial hypergraph of H induced by the hyperedges $\{e \in E(H) : e \text{ lies in some perfect matching of } H\}$ is connected.

We split the proof of Theorem 4.22 into several parts. The equivalence of (a), (b) and (c) follows from results in [Scheidweiler, 2011], the equivalence of the other statements is new to the best of our knowledge. We first explain how "(a) $\iff$ (b) $\iff$ (c) follows from results stated in [Scheidweiler, 2011]. Those are formulated in terms of d-maximum matchings and minimum d-vertex cover. As we only need the E-maximum case, we reformulate them as follows.

Corollary 4.23 (implied by Corollary 2.26 and 2.33 of [Scheidweiler, 2011])**.** *If e is a hyperedge in a balanced hypergraph H, then e is contained in an E-maximum matching if and only if $|e \cap T| = 1$ for every minimum E-vertex cover T of H.*

Corollary 4.24 (implied by Corollary 2.29 and 2.32 of [Scheidweiler, 2011])**.** *If e is a hyperedge in a balanced hypergraph H, then e is contained in a minimum hyperedge cover if and only if $e \cap S \neq \emptyset$ for every maximum stable set S of H.*

The two corollaries above imply the equivalence of (a), (b) and (c).

Corollary 4.25. *If H is a balanced, r-uniform hypergraph, then H is 1-extendable if and only if every minimum E-vertex cover is a maximum stable set, which is the case if and only if every maximum stable set is a minimum E-vertex cover.*

Proof. If H is matching covered and r-uniform, then the maximum size of a matching, the minimum size of an E-vertex cover, the minimum size of a hyperedge cover, and the maximum size of a stable set are all equal to $|V(H)|/r$. Furthermore, every maximum size matching is a minimum size hyperedge cover, and vice versa. For all $e \in E(H)$ we have $|e \cap T| = 1$ for every minimum E-vertex cover T of H by Corollary 4.23, and $e \cap S \neq \emptyset$ for every maximum stable set S of H by Corollary 4.24. Thus, every minimum E-vertex cover is a maximum stable set, and every maximum stable set is a minimum E-vertex cover.

Now, suppose every minimum E-vertex cover is a maximum size stable set. The size of a minimum E-vertex cover equals the maximum size of a matching in H, which is at most $|V(H)|/r$. On the other hand, the size of a maximum stable set is equal to the minimum size of a hyperedge cover, which is at least $|V(H)|/r$. Thus, the minimum size of an E-vertex cover is equal to $|V(H)|/r$. By Corollary 4.23 and the assumption that every minimum E-vertex cover is stable, every hyperedge is contained in an E-maximum matching. Every E-maximum matching has size $|V(H)|/r$ and is therefore a perfect matching. Thus, H is matching covered.

Finally, suppose every maximum stable set is a minimum E-vertex cover. As above, we get that the size of a maximum stable set is equal to $|V(H)|/r$. By duality, also the size of a minimum hyperedge cover is $|V(H)|/r$, which implies that every hyperedge cover is a perfect matching. By Corollary 4.24, every hyperedge is contained in a perfect matching. □

As a next step we prove a Hall-type theorem for 1-extendability, namely, we show that (a) and (d) are equivalent. Note that the first two conditions of (d) are similar to that of Theorem 3.18 in the case $f(v) = 1$ for all vertices v.

Lemma 4.26. *Let H be a balanced, r-uniform hypergraph, and $V_1, \dots, V_r$ be an r-partition of $V(H)$. The hypergraph H is 1-extendable if and only if*

(1) $|V_i| = |V_j|$ for all $i, j \in \{1, \dots, r\}$,

(2) $|X| \leq |Y|$ *for all* $X \subseteq V_1$, $Y \subseteq V(H) \setminus V_1$, $Y \in \mathcal{N}(X)$ *with* $(V_1 \setminus X) \cup Y$ *a stable set, and*

(3) $|X| + 1 \leq |Y|$ *for all* $X \subseteq V_1$, $Y \subseteq V(H) \setminus V_1$, $Y \in \mathcal{N}(X)$ *with* $(V_1 \setminus X) \cup Y$ *not a stable set.*

Proof. If H is matching covered, then it has a perfect matching and the first two statements follow from Theorem 3.18. Let $X \subseteq V_1$, $Y \subseteq V(H) \setminus V_1$, $Y \in \mathcal{N}(X)$ with $(V_1 \setminus X) \cup Y$ not a stable set. We have to show that $|X| + 1 \leq |Y|$. As $(V_1 \setminus X) \cup Y$ is not stable, there exists a hyperedge $e^* \in E(H)$ intersecting this set in at least two vertices. Let M be a perfect matching containing e^*, which must exist as H is matching covered. First, we consider the case $|e^* \cap (V_1 \setminus X)| = 0$ and $|e^* \cap Y| \geq 2$. This means that $|e^* \cap X| = 1 \leq |e^* \cap Y| - 1$. For all other $e \in M$ we have $|e \cap X| \leq |e \cap Y|$ because $Y \in \mathcal{N}(X)$ and $X \subseteq V_1$. If $|e^* \cap (V_1 \setminus X)| = 1$ and $|e^* \cap Y| \geq 1$, then $|e^* \cap X| = 0$, and thus $|e^* \cap X| \leq |e^* \cap Y| - 1$. In both cases we obtain

$$|X| = \sum_{e \in M} |e \cap X| \leq \sum_{e \in M} |e \cap Y| - 1 = |Y| - 1.$$

Now, assume that the three conditions stated above hold. By Theorem 3.18, H has a perfect matching. Suppose that there exists a hyperedge $e^* \in E(H)$ not contained in a perfect matching. By Corollary 3.6 in [Scheidweiler, 2011] there exists a minimum E-vertex cover T of H intersecting e^* in more than one vertex. We set $X = V_1 \setminus T$ and $Y = T \setminus V_1$. For every hyperedge $e \in E(H)$ with $e \cap X \neq \emptyset$ we have $e \cap (T \cap V_1) = \emptyset$, and thus $e \cap (T \setminus V_1) \neq \emptyset$ as T is an E-vertex cover. This implies that there exists $Y' \subseteq Y$ with $Y' \in \mathcal{N}(X)$. If Y' is a strict subset of Y, then $(V_1 \cap T) \cup Y'$ is an E-vertex cover of smaller size than T. This is impossible as T is a minimum E-vertex cover. Thus, $Y \in \mathcal{N}(X)$. Observe that $(V_1 \setminus X) \cup Y = T$ is not a stable set, so by the third condition of the lemma we get $|X| + 1 \leq |Y|$, which is equivalent to $|V_1| + 1 \leq |T|$. But V_1 and T are both minimum E-vertex covers, and thus we obtain a contradiction. □

Finally, we show that (e) and (a) are equivalent. This completes our proof of the various characterizations of 1-extendable, uniform, balanced hypergraphs given in Theorem 4.22.

Lemma 4.27. *A balanced, r-uniform hypergraph H on more than r vertices is matching covered if and only if $H - F$ has a perfect matching for every $F \subseteq V(H)$ of size r with $|F \cap T| = 1$ for all stable E-vertex covers T.*

Proof. First, we assume that the "only if"-condition holds. For every hyperedge $e \in E(H)$ we have $|e| = r$, and $|e \cap T| = 1$ for all stable E-vertex covers T of H.

Thus, $H[V(H) \setminus e]$ has a perfect matching M_e for every $e \in E(H)$, and $M_e \cup \{e\}$ is a perfect matching of H containing e.

Now, assume H is matching covered, and let $F \subseteq V(H)$ be as stated above. We look at the hypergraph $H \setminus F$, which is the subhypergraph of H restricted to $V(H) \setminus F$. Every stable set S of $H \setminus F$ is a stable set of H with $S \cap F = \emptyset$. Every maximum stable set of H is an E-vertex cover and therefore has a non-empty intersection with F. This implies that the maximum size of a stable set in $H \setminus F$ is less than the maximum size of a stable set in H, which is ${|V(H)|}/{r}$.

Now, let $R \subseteq E(H)$ be a minimum size hyperedge cover of $H \setminus F$. A trivial lower bound on the size of R is ${|V(H \setminus F)|}/{r}$, which is equal to ${|V(H)|}/{r} - 1$. On the other hand, the minimum size of a hyperedge cover in $H \setminus F$ is equal to the maximum size of a stable set in $H \setminus F$ because $H \setminus F$ is a balanced hypergraph. Together, it follows that R contains exactly ${|V(H) \setminus F|}/{r}$ hyperedges. The hyperedges of $H \setminus F$ are of the form $e \setminus F$. As $|R| = {|V(H \setminus F)|}/{r}$, the set R uses only hyperedges of size r and those hyperedges are pairwise disjoint. Therefore, R contains only hyperedges $e \in E(H)$ with $e \cap F = \emptyset$, and R forms a perfect matching of $H \setminus F$. In particular, R is also a perfect matching of $H - F$. □

Now, we turn to the relation between k-extendability and duality in balanced hypergraphs. If H is a balanced, r-uniform hypergraph, then its dual H^* is balanced and every vertex has degree r. This implies that the hyperedge set of H^* is the disjoint union of r perfect matchings because a balanced hypergraph is normal. In particular, H^* is matching covered. If we further assume that H has a perfect matching, we are able to characterize when H is k-extendable in terms of its dual hypergraph.

Lemma 4.28. *Let H be a balanced, r-uniform, 1-extendable hypergraph, and k be a natural number with $2 \leq k \leq {|V(H)|}/{r}$. If H is k-extendable, then H^* has no maximal matching of size ${|V(H)|}/{r} - k + 1$.*

Proof. Suppose that H^* has a maximal matching of size ${|V(H)|}/{r} - k + 1$, then this matching corresponds to a maximal stable set S in H. The set S is not a vertex cover of H because the minimum size of a vertex cover in H is equal to the maximum size of a matching, which is ${|V(H)|}/{r}$ as H is an r-uniform hypergraph with a perfect matching. Let $F := \{e \in E(H) : e \cap S = \emptyset\}$ be the set of hyperedges of H not covered by S.

First, we consider the case that $H[F]$ has a matching M' of size k. Let M be a perfect matching of H containing M', which exists as H is k-extendable. It follows that

$$|S| = \sum_{e \in M} |e \cap S| = |M| - |M' \cap M| = \frac{|V(H)|}{r} - k,$$

contradicting $|S| = {|V(H)|}/{r} - k + 1$.

Next, we consider the case $\nu_E(H[F]) \leq k-1$. For any perfect matching M we have $|S| = |M| - |M \cap F|$, in particular, $|M \cap F| = k-1$ for any perfect matching M of H, and thus $\nu_E(H[F]) = k-1$. As the partial hypergraph $H[F]$ is also balanced, it has a minimum E-vertex cover T of size $k-1$. We claim that $F = \bigcup_{v \in T} \delta_H(v)$. Otherwise, there exists a vertex $w \in T$, and a hyperedge $\tilde{e} \in \delta_H(w) \setminus F$. Let $\widetilde{M}$ be a perfect matching containing $\tilde{e}$. We know that $|\widetilde{M} \cap F| = k-1$, and every $f \in \widetilde{M} \cap F$ is covered by some $v \in T$. But then T covers k disjoint hyperedges, namely $\left(\widetilde{M} \cap F\right) \cup \{\tilde{e}\}$. This is not possible because $|T| = k-1$. Thus, $\delta_H(v) \subseteq F$ for all $v \in T$.

The set $S' := S \cup \{w\}$ for any $w \in T$ is a stable set of H because $|e \cap S'| = |e \cap S| = 1$ for all $e \in E(H) \setminus F$ and $|e \cap S'| = |e \cap \{w\}| \leq 1$ for all $e \in F$. Thus, S is not a maximal stable set. □

One problem that arises when trying to prove a converse of Lemma 4.28 is that a k-extendable hypergraph might not be $(k-1)$-extendable. Thus, we only characterize when a hypergraph is simultaneously 1-extendable, 2-extendable, ..., k-extendable.

Lemma 4.29. *Let H be a balanced, r-uniform, 1-extendable hypergraph, and k be a natural number with $2 \leq k \leq \frac{|V(H)|}{r}$. The hypergraph H is k'-extendable for every $k' = 2, \ldots, k$ if and only if H^* has no maximal matching S such that $|V(H)|/r - k + 1 \leq |S| \leq |V(H)|/r - 1$.*

Proof. By Lemma 4.28, H^* has no maximal matching of size $|V(H)|/r - k' + 1$ if H is k'-extendable. It remains to show that if H^* has no maximal matching S with $|V(H)|/r - k + 1 \leq |S| \leq |V(H)|/r - 1$, then H is k'-extendable for every number $k' = 2, \ldots, k$. Suppose that H has a matching of size at most k not contained in any perfect matching, and let M' be the smallest such matching. Let $f \in M'$ be a fixed hyperedge of M', and consider the hypergraph $H' := H - \bigcup_{e \in M' \setminus \{f\}} e$. This hypergraph has a perfect matching because $M' \setminus \{f\}$ is extendable to a perfect matching in H by the choice of M'. However, H' has no perfect matching containing f. Otherwise, such a perfect matching together with $M' \setminus \{f\}$ would form a perfect matching of H containing M'. By Corollary 4.24, there exists a maximum stable set S' of H' with $f \cap S' = \emptyset$. Let S be a maximal stable set of H containing S'. Every vertex $v \in S \setminus S'$ is contained in a hyperedge $e \in M' \setminus \{f\}$. Thus, S does not intersect f, and therefore it is not an E-vertex cover. Thus, $|S| \leq |V(H)|/r - 1$. On the other hand, $|S| \geq |S'| = \nu_E(H') = |V(H')|/r = |V(H)|/r - |M'| + 1 \geq |V(H)|/r - k + 1$. This implies that S corresponds to a maximal matching in the dual hypergraph H^* such that $|V(H)|/r - k + 1 \leq |S| \leq |V(H)|/r - 1$. □

Now, we generalize Lemma 4.27 to the k-extendable case.

Corollary 4.30. *Let H be an r-uniform, balanced, and 1-extendable hypergraph. The following statements are equivalent for any $k \in \mathbb{N}$ with $2 \leq k \leq |V(H)|/r$.*

(a) *H is k'-extendable for $k' = 2, \ldots, k$.*

(b) *For every $k' \in \{2, \ldots, k\}$ we have that $H - F$ has a perfect matching for every set $F \subseteq V(H)$ of size $r \cdot k'$ with $|F \cap T| = k'$ for all stable E-vertex covers T of H.*

(c) *For every $k' \in \{2, \ldots, k\}$ we have that $H - (F_1 \cup \ldots \cup F_{k'})$ has a perfect matching for every choice of k' pairwise disjoint sets $F_1, \ldots, F_{k'} \subseteq V(H)$, where each set has size r and $|F_i \cap T| = 1$ for all stable E-vertex covers T of H and all $i \in [k']$.*

Proof. (b) implies (c): If $F_1, \ldots, F_{k'}$ are disjoint sets of size r with $|F_i \cap T| = 1$ for all stable E-vertex covers T of H and all $i \in [k']$, then $F = F_1 \cup \ldots \cup F_{k'}$ has size $r \cdot k'$ and satisfies $|F \cap T| = k'$ for all stable E-vertex covers T.

(c) implies (a): If $\{e_1, \ldots, e_{k'}\}$ is a matching of size k' with $2 \leq k' \leq k$, then $|e_i \cap T| = 1$ for every stable E-vertex cover T, and thus $H - (e_1 \cup \ldots \cup e_{k'})$ has a perfect matching M. Then, $M \cup \{e_1, \ldots, e_{k'}\}$ is a perfect matching of H. Thus, every matching of size at most k in H can be extended to a perfect matching.

(a) implies (b): Let H be a hypergraph that is k'-extendable for $k' = 1, \ldots, k$, and $F \subseteq V(H)$ a set of size $r \cdot k'$ with $|F \cap T| = k'$ for all stable E-vertex covers T of H. We consider the hypergraph $H \setminus F$, which consists of all hyperedges of the form $e \setminus F$ if $e \nsubseteq F$. Let S be a maximum stable set of $H \setminus F$. The set S is also a stable set of H that might not be maximal. Let S' be a maximal stable set of H containing S. If $|S' \setminus S| \geq k'$, then $|S| \leq {}^{|V(H)|}/_{r} - k'$. Otherwise, $|S' \cap F| = |\,(S' \setminus S) \cap F| < k'$ and S' is not a stable E-vertex cover of H because every stable E-vertex cover intersects F in exactly k vertices by assumption. By Lemma 4.29, $|S'| \leq {}^{|V(H)|}/_{r} - k'$, and thus $|S| \leq {}^{|V(H)|}/_{r} - k' + k' - 1 = {}^{|V(H)|}/_{r} - 1$. Again by Lemma 4.29 it follows that $|S| \leq {}^{|V(H)|}/_{r} - k$.

As $H \setminus F$ is balanced, the size of S gives an upper bound on the size of a minimum hyperedge cover. A trivial lower bound is ${}^{|V(H \setminus F)|}/_{r}$, which is equal to ${}^{|V(H)|}/_{r} - k'$. This implies that a minimum hyperedge cover of $H \setminus F$ has size ${}^{|V(H \setminus F)|}/_{r}$, and is therefore a perfect matching of $H \setminus F$ using only hyperedges of size r. Such a hyperedge cover forms also a perfect matching of $H - F$. □

An open question is whether one can decide in polynomial time if a balanced hypergraph is k-extendable. For constant k this is clearly possible. Given a balanced hypergraph H, and a constant k with $1 \leq k \leq \nu_E(H)$, we first test whether H has a perfect matching, which can be done in polynomial time using linear programming. If H has no perfect matching it is not k-extendable. Otherwise, we look at all k-element subsets F of $E(H)$. If F is a matching in H, we test whether $H - F$ has a perfect matching. If $H - F$ has a perfect matching for all k-element subsets F of $E(H)$, then H is k-extendable, otherwise it is not k-extendable. We can test in polynomial time whether a set F of hyperedges forms a matching and whether $H-F$ has a perfect matching because $H-F$ is balanced. The number of k-element subsets of $E(H)$ is $\mathcal{O}(|E(H)|^k)$. In total, we get a polynomial time algorithm to decide whether a balanced hypergraph is k-extendable or not if k is constant. We do not know the complexity of this problem when k is part of the input.

4.2.3 Greedily Matchable Hypergraphs

Recall that a hypergraph is called *greedily matchable* if every matching can be extended to a perfect matching. Caro, Sebő, and Tarsi give with Theorem 4.6 a characterization of greedily matchable hypergraphs in terms of forbidden induced subhypergraphs. However, the concrete structure of greedily matchable hypergraphs is not known. It seems that it is much more difficult to characterize them than greedily matchable graphs. Namely, a graph G with a perfect matching is greedily matchable if and only if it is $(|V(G)|/2-1)$-extendable by Theorem 4.1. This means that the question whether a graph is greedily matchable reduces to the problem whether it is k-extendable for one specific value of k. This result does not carry over to hypergraphs. There are r-uniform hypergraphs H that are $({}^{|V(H)|}/_{r}-1)$-extendable but not greedily matchable, for example the hypergraphs $H_{r,k}$ described in Example 4.17. In this subsection we give some examples of greedily matchable hypergraphs, describe constructions that build new greedily matchable hypergraphs from old ones, characterize 2-regular greedily matchable hypergraphs, and show that one can decide in polynomial time whether a hypergraph of maximum degree bounded by some constant is greedily matchable.

The only greedily matchable graphs are the complete bipartite graph and the complete graph on an even number of vertices. We give two classes of hypergraphs generalizing these graphs.

Definition 4.31 (complete r-partite, complete r-uniform hypergraph)**.** Given integers $n, r \in \mathbb{N}$ the *complete r-partite hypergraph* K_{n^r} has vertex set $V_1 \cup \ldots \cup V_r$ where $V_i = \{v_{i,1}, \ldots, v_{i,n}\}$ for $i \in [r]$, and hyperedges $\{v_{1,j_1}, \ldots, v_{r,j_r}\}$ for every $v_{i,j_i} \in V_i$, $i \in [r]$. The *complete r-uniform hypergraph* K_n^r on n vertices has vertex set $\{v_1, \ldots, v_n\}$ and its hyperedge set consists of all r-element subsets of $V(K_n^r)$.

Clearly, all maximal matchings of K_{n^r} and K_n^r have size n and $\lfloor \frac{n}{r} \rfloor$, respectively. In particular, K_{n^r} is always greedily matchable, and K_n^r is greedily matchable if n is divisible by r.

Observation 4.32. *The hypergraphs K_{n^r} and K_{rn}^r are greedily matchable for all $r, n \in \mathbb{N}$.*

In the case $r=2$ the hypergraphs K_{n^r} and K_{rn}^r are the only connected, greedily matchable instances. However, for higher ranks there are more classes of greedily matchable hypergraphs. For example, using the vertex multiplication operation (see Definition 2.33) from Chapter 2 we can construct new greedily matchable hypergraphs from known ones.

Observation 4.33. *Let H be a hypergraph and $c : V(H) \to \mathbb{N}$ be a function on its vertex set. The hypergraph H is greedily matchable if and only if $H^{(c)}$ is greedily matchable.*

Proof. Every hyperedge of $E(H^{(c)})$ corresponds to a hyperedge of $E(H)$, and vice versa. If we denote the hyperedges of $E(H^{(c)})$ by $e^{(c)}$, where $e \in E(H)$ is such that $e^{(c)} = \{v^{(i)} : v \in e, 1 \leq i \leq c(v)\}$, then two hyperedges $e^{(c)}$ and $f^{(c)}$ in $H^{(c)}$ have a non-empty intersection if and only if $e, f \in E(H)$ have a non-empty intersection as $c(v) \geq 1$ for all $v \in V(H)$. Thus, the matchings in $H^{(c)}$ correspond one to one to the matchings in H, and a matching in $H^{(c)}$ is perfect if and only if the corresponding matching in H is perfect. This implies that $H^{(c)}$ is greedily matchable if and only if H is greedily matchable. □

Using the previous observation we can construct connected, greedily matchable hypergraphs that are not isomorphic to K_{n^r} or K^r_{rn}. For example, $K^{(2)}_{2,2}$ (every vertex of $K_{2,2}$ gets doubled) is a connected, 4-partite, greedily matchable hypergraph with eight vertices and four hyperedges, thus it cannot be isomorphic to a complete 4-partite hypergraph.

One might conjecture that all connected, greedily matchable hypergraphs are multiplications of K_{n^r} or K^r_{rn} for some $r, n \in \mathbb{N}$. This is not true. There is another construction that preserves the property of being greedily matchable. It works as follows:

Let H_1, H_2 be two greedily matchable hypergraphs with $V(H_1) \cap V(H_2) = \emptyset$, and $v_1^* \in V(H_1), v_2^* \in V(H_2)$ be two fixed vertices. We define $H = H_1 \oplus H_2$ as follows. Its vertex set is $V(H) = V(H_1) \cup V(H_2) \cup \{z_1, z_2\}$, where $z_1, z_2 \notin V(H_1) \cup V(H_2)$ are two new vertices. It contains all hyperedges $e \in E(H_1) \setminus \delta_{H_1}(v_1^*)$, all $e \in E(H_2) \setminus \delta_{H_2}(v_2^*)$, and for each $e \in \delta_{H_1}(v_1^*) \cup \delta_{H_2}(v_2^*)$ it contains the two hyperedges $e \cup \{z_1\}$ and $e \cup \{z_2\}$.

Lemma 4.34. *If H_1, H_2 are greedily matchable, then $H = H_1 \oplus H_2$ is greedily matchable.*

Proof. If M is a matching of H, then $M_1 = M \cap E(H_1)$ is a matching of H_1 and $M_2 = M \cap E(H_2)$ one of H_2. If M covers v_1^*, then it contains a hyperedge $m_1 = e_1 \cup \{z_i\}$ with $i = 1$ or $i = 2$, $e_1 \in E(H_1)$ and $v_1^* \in e_1$. In this case, also $M_1 \cup \{e_1\}$ is a matching of H_1 and we replace M_1 by $M_1 \cup \{e_1\}$. Similar, if M covers v_2^*, then it contains a hyperedge $m_2 = e_2 \cup \{z_i\}$ with $i = 1$ or $i = 2$, $e_2 \in E(H_2)$, $v_2^* \in e_2$, and we replace M_2 by $M_2 \cup \{e_2\}$. As H_1 and H_2 are greedily matchable, there exist a perfect matching $\widetilde{M}_1$ of H_1 and a perfect matching $\widetilde{M}_2$ of H_2 with $M_1 \subseteq \widetilde{M}_1$ and $M_2 \subseteq \widetilde{M}_2$. If M does not cover v_1^*, then at least one

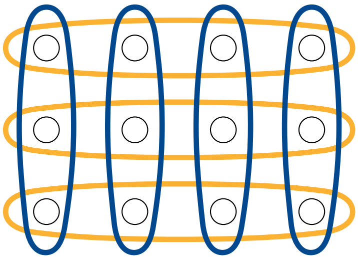


Figure 4.7: The hypergrid $D_{3,4}$, where the two disjoint perfect matchings are drawn in orange and blue.

of z_1, z_2 is missed by M. Let $e_1 \in \widetilde{M}_1$ be the hyperedge of $\widetilde{M}_1$ covering v_1^*, then $m_1 = e_1 \cup \{z_1\}$ or $m_1 = e_2 \cup \{z_2\}$ is a hyperedge disjoint from the hyperedges of M. We define m_2 similarly if M did not already cover v_2^*. In total, we get that $\widetilde{M} = \{e \in M_1 : v_1^* \notin e\} \cup \{e \in M_2 : v_2^* \notin e\} \cup \{m_1, m_2\}$ is a perfect matching of H containing M. □

The previous lemma implies that there are arbitrarily large greedily matchable hypergraphs that are not 3-vertex connected. This is a sharp contrast to the graph case where only K_2 and $K_{2,2}$ are not 3-connected. Every greedily matchable graph on $2n$ vertices is at least n-vertex connected, i.e., greedily matchable graphs are highly connected.

Now, we characterize greedily matchable, 2-regular hypergraphs. If a greedily matchable, connected hypergraph H has a vertex of degree one, then $|E(H)| = 1$ because H is connected and every hyperedge is contained in a perfect matching. Thus, every 1-regular greedily matchable hypergraph consists just of one hyperedge, and greedily matchable, 2-regular hypergraphs are the first non-trivial class of greedily matchable hypergraphs. A class of greedily matchable, 2-regular hypergraphs is given in the following definition.

Definition 4.35 (Hypergrid)**.** The hypergrid $D_{r,s}$ of size $r \cdot s$ for integers $r, s \in \mathbb{N}$ is the hypergraph on the vertex set $\{v_{ij} : i \in [r], j \in [s]\}$ with hyperedges defined by $e(i, \cdot) := \{v_{i1}, \ldots, v_{is}\}$ for $i \in [r]$, as well as $e(\cdot, j) := \{v_{1j}, \ldots, v_{rj}\}$ for $j \in [s]$.

Every hypergrid $D_{r,s}$ has exactly two perfect matchings $M_1 := \{e(i, \cdot) : i \in [r]\}$ and $M_2 := \{e(\cdot, j) : j \in [s]\}$. Every hyperedge of M_1 intersects all hyperedges of M_2 and vice versa, thus every non-empty matching is a subset of either M_1 or M_2. This implies that $D_{r,s}$ is greedily matchable.

Figure 4.7 shows the hypergrid $D_{3,4}$, where the perfect matching M_1 is drawn in orange and M_2 in blue.

We show that greedily matchable, 2-regular hypergraphs are multiplications of hypergrids. Therefore, we use Theorem 4.7, which implies that a bipartite graph is a general partition graph if and only if it is a complete bipartite graph.

Theorem 4.36. *If H is a greedily matchable, 2-regular hypergraph, then there exist integer $r, s \in \mathbb{N}$ such that H is isomorphic to $D_{r,s}^{(c)}$ for some $c : V(D_{r,s}) \to \mathbb{N}$.*

Proof. If H is greedily matchable and 2-regular, then $E(H)$ is the disjoint union of two perfect matchings M_1 and M_2. This implies that the line graph $L(H)$ is a bipartite, general partition graph. By Theorem 4.7, $L(H)$ is isomorphic to $K_{r,s}$ where r is the size of M_1 and s the size of M_2. In particular, every hyperedge of M_1 has a non-empty intersection with every hyperedge of M_2. Set $M_1 = \{e_1, \dots, e_r\}$ and $M_2 = \{f_1, \dots, f_s\}$. For each $i \in [r]$ and $j \in [s]$ we have $|e_i \cap f_j| \geq 1$. In total, H is isomorphic to a multiplication of $D_{r,s}$ where v_{ij} is multiplied $|e_i \cap f_j|$-times for $i \in [r]$, $j \in [s]$. □

If H is a hypergraph with hyperedges of size at most some constant, then one can decide in polynomial time whether H is greedily matchable using Theorem 4.6. We show that one can also decide in polynomial time whether a hypergraph of bounded maximum degree is greedily matchable. First, we prove that it suffices to check whether a hypergraph is k-extendable for k up to the maximum degree in order to decide whether a hypergraph is greedily matchable.

Lemma 4.37. *A hypergraph H is greedily matchable if and only if it is k-extendable for all $k \in \mathbb{N}$ with $1 \leq k \leq \min(\Delta(H), \nu_E(H))$.*

Proof. If H is greedily matchable, then it is k-extendable for every natural number k where $1 \leq k \leq \nu_E(H)$.

For the other direction, suppose that H is not greedily matchable. Let M be any maximal matching that is not perfect. If $|M| \leq \min(\Delta(H), \nu_E(H))$, then H is not k-extendable for $k = |M|$. Otherwise, let w be a vertex of H exposed by M. For every $e \in \delta_H(w)$ there exists a hyperedge $m_e \in M$ with $m_e \cap e \neq \emptyset$. The set $\{m_e : e \in \delta_H(w)\}$ is a matching of size at most $|\delta_H(w)|$ not contained in any perfect matching. Thus, H is not k-extendable for some $k \in \{1, \dots, \Delta(H)\}$. □

Using similar arguments as in the previous lemma, we show how to decide in polynomial time whether or not a hypergraph is greedily matchable if its maximum degree is bounded by some constant.

Theorem 4.38. *For every fixed $d \in \mathbb{N}$ there exists a polynomial time algorithm that decides whether a hypergraph of maximum degree at most d is greedily matchable.*

Proof. We claim that a hypergraph H of maximum degree at most d is not greedily matchable if and only if one can find a vertex $w \in V(H)$ and a matching M of size at most d such that for every $e \in \delta_H(w)$ there exists $m_e \in M$ with $e \cap m_e \neq \emptyset$ and M does not cover w. If such a matching M exists, then M cannot be contained in a matching covering w, and thus M is not contained in a perfect matching. On the other hand, if H is not greedily matchable, then there exists a maximal matching M that is not perfect. Let w be a vertex not covered by M. For every $e \in \delta_H(w)$ there exists a hyperedge $m_e \in M$ with $e \cap m_e \neq \emptyset$. Now, the set $\{m_e : e \in \delta_H(w)\}$ is a matching of size at most d with the desired properties.

As a next step we show that we can test in polynomial time whether H has a vertex w and a matching M of size at most d with the properties that w is not covered by M, and for every $e \in \delta_H(w)$ there exists $m_e \in M$ with $e \cap m_e \neq \emptyset$. For a fixed vertex $w \in V(H)$ and a matching M of size at most d we can check these two properties in constant time as $\delta_H(w) \leq d$ and d is constant. As we can enumerate all matchings of size at most d in $\mathcal{O}(|E(H)|^d)$-time, we can decide in $\mathcal{O}(|V(H)| \cdot |E(H)|^d)$-time whether or not H is greedily matchable. □

The complexity of deciding whether a hypergraph is greedily matchable remains open if neither its maximum degree nor its rank is bounded by some constant. It might be that the problem becomes easier on balanced hypergraphs by utilizing the connection between extendability in a hypergraph and maximal matchings in its dual given in Lemma 4.29.

4.3 Matching Covered Hypergraphs and Tight Cuts

In this section we define tight cuts and tight cut contractions in hypergraphs and investigate their properties. In particular, we show that there is a one-to-one correspondence between perfect matchings in a hypergraph and pairs of perfect matchings in the two hypergraphs obtained by contracting the shores of a tight cut. Additionally, we investigate basic properties of tight cuts and tight cut contractions in the first subsection.

In the second subsection we look at the relation between tight cuts and the perfect matching polytope. As in the graph case, every tight cut yields a decomposition of the perfect matching polytope of a hypergraph. This result is used to prove that the tight cut contractions of a balanced uniform hypergraph remain balanced. Furthermore, we investigate separating cuts in hypergraphs and show that the existence of a non-tight separating cut implies that the perfect matching polytope is not given by the degree and non-negativity constraints.

4.3.1 Basic Properties

We start with a formal definition of tight cuts in hypergraphs, which is literally the same as for graphs.

Definition 4.39 (Tight cut)**.** Let H by a hypergraph with a perfect matching and $A \subseteq V(H)$ be a set of vertices. The cut $\delta_H(A)$ is *tight* if every perfect matching contains exactly one hyperedge of it. A tight cut $\delta_H(A)$ is called *trivial* if $|A| = 1$ or $|V(H) \setminus A| = 1$, otherwise it is called *non-trivial.*

In contrast to the graph case we have to remember the set of vertices defining a cut and cannot only consider a cut as a set of edges. In a connected graph a cut is defined by a set of vertices and its complement but not by any other set. For connected hypergraphs this is not true anymore. Namely, it is possible that $\delta_H(S)$ and $\delta_H(T)$ coincide even if $T \neq S$ and $T \neq V(H) \setminus S$. It even might happen that a set A and a vertex v define the same cut for some $v \in V(H)$ and $A \subseteq V(H)$ with $|A| > 1$ and $|V(H) \setminus A| > 1$.

Tight cuts can be used to decompose a hypergraph into smaller ones as this is done in the graph case. We denote by $\overline{A}$ the complement of a set A if the ground set is clear from the context.

Definition 4.40 (Tight cut contraction)**.** Let H be a hypergraph with a tight cut $\delta_H(A)$. We introduce two new vertices a and $\overline{a}$ not contained in $V(H)$, and define for every hyperedge $e \in \delta_H(A)$ two new hyperedges $e_a := (e \setminus A) \cup \{a\}$ and $e_{\overline{a}} = (e \cap A) \cup \{\overline{a}\}$.

The *tight cut contractions* of H with respect to A and $\overline{A}$ are the hypergraphs H_A and $H_{\overline{A}}$ with vertex sets $V(H_A) := \{a\} \cup \overline{A}$, $V(H_{\overline{A}}) := \{\overline{a}\} \cup A$, and hyperedge sets $E(H_A) := \{e \in E(H) \colon e \subseteq \overline{A}\} \cup \{e_a \colon e \in \delta_H(A)\}$, $E(H_{\overline{A}}) := \{e \in E(H) \colon e \subseteq A\} \cup \{e_{\overline{a}} \colon e \in \delta_H(A)\}$.

Loosely speaking, H_A and $H_{\overline{A}}$ are the hypergraphs obtained from H by contracting A and $\overline{A}$. Figure 4.8 shows a non-trivial tight cut in a 3-uniform hypergraph and the associated tight cut contractions. Observe, that one of the two contractions is not 3-uniform.

When considering tight cuts in graphs, parity arguments play a huge role. In general hypergraphs we cannot use such arguments. Therefore, we consider hypergraphs that can be made uniform by vertex multiplication (see Definition 2.33). Formally, we look at hypergraphs with the following property.

Definition 4.41 (Uniformizable hypergraph)**.** A hypergraph H is called *uniformizable* if there exists a function $m : V(H) \to \mathbb{N}$ with the property that the multiplication $H^{(m)}$ is r-uniform for some $r \in \mathbb{N}$.

We show how to decide in polynomial time whether a hypergraph H is uniformizable, and, if it is, find a function $m : V(H) \to \mathbb{N}$, and an integer $r \in \mathbb{N}$ such that $H^{(m)}$ is r-uniform.

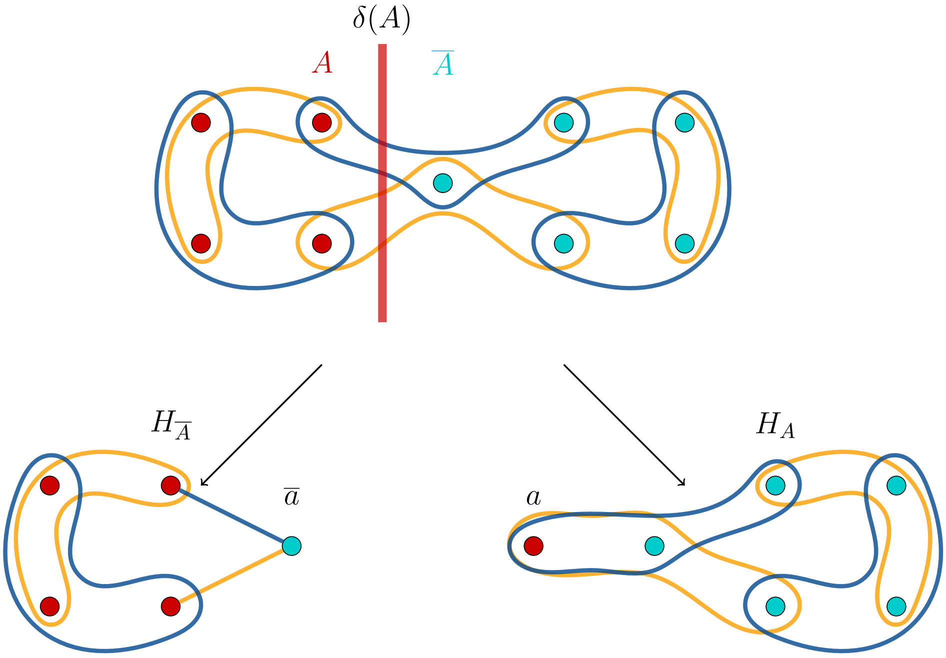


Figure 4.8: A tight cut $\delta_H(A)$ in a hypergraph H, and its tight cut contractions H_A and $H_{\overline{A}}$.

For a given hypergraph H we look at the system

$$\sum_{v \in e} m_v = r \qquad \forall e \in E(H), \tag{4.4}$$

$$m_v \geq 1 \qquad \forall v \in V(H), \tag{4.5}$$

$$r \geq 1. \tag{4.6}$$

It has an integral solution if and only if $H^{(m)}$ is r-uniform. Finding integral solutions of a system of linear inequalities is in general $\mathcal{NP}$-hard. In this special case one can find an integral solution in polynomial time. First, we compute $m^* \in \mathbb{Q}^{V(H)}$, $r^* \in \mathbb{Q}$ satisfying (4.4)-(4.6), which can be done in polynomial time. Let $d \in \mathbb{N}$ be the least common multiple of the denominators of m_v for $v \in V(H)$. The number d can also be computed in polynomial time using the Euclidean Algorithm. Finally, we define a function $m : V(H) \to \mathbb{N}$ by $m(v) := d \cdot m_v$ for all $v \in V(H)$. With this definition we have $\sum_{v \in e} m(v) = d \cdot r^*$ for all $e \in E(H)$, and $m(v) \geq d \geq 1$ for all $v \in V(H)$. This means that the system (4.4)-(4.6) has an integral solution if and only if it has a fractional solution. In total, we can decide in polynomial time whether H is uniformizable.

Clearly, every uniform hypergraph is uniformizable (set $m \equiv 1$). We do not claim that uniformizable hypergraphs have a much richer structure than uniform hypergraphs; on the contrary, uniformizable hypergraphs seem to be very similar to uniform ones. The reason why we work with uniformizable hypergraphs is that the tight cut contractions of a uniform hypergraph are only uniformizable and not uniform. To prove this we need the following lemma.

Lemma 4.42. *Let H be a matching covered, uniformizable hypergraph with a function $m : V(H) \to \mathbb{N}$ such that $H^{(m)}$ is r-uniform for some $r \in \mathbb{N}$.*

(i) If $\delta_H(A)$ is a tight cut, then $m(e \cap A) = k$ for all $e \in \delta_H(A)$, where k is the unique integer in $\{1, \dots, r-1\}$ with $k \equiv m(A)$ mod r.

(ii) If $A \subseteq V(H)$ is a set with $m(A) \equiv 0$ mod r, then $|M \cap \delta_H(A)| \geq 2$ or $|M \cap \delta_H(A)| = 0$ for all perfect matchings M of H. If $A \notin \{\emptyset, V(H)\}$ and $m(A) \equiv 0$ mod r, then there exists a perfect matching M with $|M \cap \delta_H(A)| \geq 2$.

Proof. Let $\delta_H(A)$ be a tight cut, $e' \in \delta_H(A)$ arbitrary, and M a perfect matching containing e'. The set $M_A := \{e \subseteq A : e \in M\}$ is a matching of $H[A]$ covering $A \setminus e'$. If we sum up the function m over all $v \in V(M_A)$, then we obtain $m(V(M_A)) = m(A) - m(e' \cap A)$. On the other hand, $m(V(M_A)) = \sum_{e \in M_A} \sum_{v \in e} m(v) = |M_A| \cdot r$. Together, we get $m(A) \equiv m(e' \cap A)$ mod r. As $1 \leq m(e' \cap A) \leq r - 1$ the claim follows.

For the second part of the observation, let $A \subseteq V(H)$. If there exists a perfect matching M intersecting the cut $\delta_H(A)$ in exactly one hyperedge e', then $\sum_{v \in e' \cap A} m(v) \equiv \sum_{v \in A} m(v) \equiv 0$ mod r follows. As $m \geq 1$ and $\sum_{v \in e'} m(v) = r$, this implies that $e' \cap A = \emptyset$, contradicting $e' \in \delta_H(A)$. Thus, if M is a perfect matching intersecting $\delta_H(A)$, then $|M \cap \delta_H(A)| \geq 2$. Now, if $A \notin \{\emptyset, V(H)\}$, then $\delta_H(A) \neq \emptyset$ as H is assumed to be connected. The hypergraph H is matching covered, which implies that for every $e' \in \delta_H(A)$ there exists a perfect matching containing e'. Every such perfect matching M intersects $\delta_H(A)$ and therefore satisfies $|M \cap \delta_H(A)| \geq 2$. □

The first observation of Lemma 4.42 implies that tight cut contractions of a uniformizable hypergraph are uniformizable.

Observation 4.43. *If H is a matching covered, uniformizable hypergraph with a tight cut $\delta_H(A)$, then H_A and $H_{\overline{A}}$ are uniformizable.*

Proof. Let $m : V(H) \to \mathbb{N}$ be a function such that $H^{(m)}$ is r-uniform.

By Lemma 4.42, $m(e \cap A) = k$ for all $e \in \delta_H(A)$ where $k \in \{1, \dots, r-1\}$ with $k \equiv m(A)$ mod r. If we define $m_{\overline{A}} : V(H_{\overline{A}}) \to \mathbb{N}$ by $m_{\overline{A}}(v) = m(v)$ for all $v \in A$, and $m_{\overline{A}}(\overline{a}) = r - k$, then $\sum_{v \in e} m_{\overline{A}}(v) = r$ for all hyperedges $e \subseteq A$, and $\sum_{v \in e_{\overline{a}}} m_{\overline{A}}(v) = \sum_{v \in e \cap A} m(v) + r - k = r$. This shows that $H_{\overline{A}}^{(m_{\overline{A}})}$ is r-uniform.

Similarly, if we define $m_A : V(H_A) \to \mathbb{N}$ by $m_A(v) = m(v)$ for all $v \in V(H) \setminus A$ and $m_A(a) = k$, then $H_A^{(m_A)}$ is r-uniform. □

Tight cut contractions are useful as there is a correspondence between perfect matchings in a hypergraph and certain pairs of perfect matchings in its tight cut contractions. In particular, the property of being matching covered is preserved under tight cut contractions. Here, we do not need our hypergraphs to be uniformizable.

Theorem 4.44. *If H is a hypergraph with a perfect matching and $A \subseteq V(H)$ is a set of vertices defining a tight cut $\delta_H(A)$, then H is matching covered if and only if H_A and $H_{\overline{A}}$ are matching covered.*

Proof. We say that a perfect matching M_A of H_A agrees with a perfect matching $M_{\overline{A}}$ of $H_{\overline{A}}$ on $\delta_H(A)$ if there exists a hyperedge $m \in \delta_H(A)$ such that m_a is the unique hyperedge in $M_A \cap \delta_{H_A}(A)$ and $m_{\overline{a}}$ is the unique hyperedge in $M_{\overline{A}} \cap \delta_{H_{\overline{A}}}(\overline{a})$. In this case, $M := M_A \cup M_{\overline{A}} \cup \{m\} \setminus \{m_a, m_{\overline{a}}\}$ is a perfect matching of H because it covers the vertices $V(M_A \setminus \{m_a\}) \cup V(M_{\overline{A}} \setminus \{m_{\overline{a}}\}) \cup m = \left(\overline{A} \setminus m\right) \cup (A \setminus m) \cup m = V(H)$.

On the other hand, let M be a perfect matching in H, we define two sets $M_A := \{e \in M \colon e \subseteq \overline{A}\} \cup \{m_a\}$, and $M_{\overline{A}} := \{e \in M \colon e \subseteq A\} \cup \{m_{\overline{a}}\}$, where m is the unique hyperedge in $M \cap \delta_H(A)$. It is straightforward to verify that M_A and $M_{\overline{A}}$ are perfect matchings of H_A and $H_{\overline{A}}$ agreeing on $\delta_H(A)$.

The correspondence between perfect matchings in H and pairs of perfect matchings in H_A, $H_{\overline{A}}$ agreeing on $\delta_H(A)$ implies that H is matching covered if and only if H_A and $H_{\overline{A}}$ are matching covered. □

4.3.2 Tight Cuts and the Perfect Matching Polytope

In this subsection we compare the perfect matching polytopes of matching covered hypergraphs and their tight cut contractions. Loosely speaking, every tight cut of a hypergraph decomposes the perfect matching polytope of a hypergraph into those of the corresponding tight cut contractions. Using this connection, we prove that tight cut contractions of balanced uniformizable hypergraphs remain balanced. Furthermore, we consider another class of cuts: the so-called separating cuts. We show that the existence of a separating cut that is not tight in a uniformizable hypergraph implies that the fractional perfect matching polytope is not integral.

For a hypergraph H we denote by $\mathcal{P}_{\mathrm{PM}}(H)$ its perfect matching polytope, i.e., the convex hull of the incidence vectors of perfect matchings. To describe the relation between the perfect matching polytope of a hypergraph H and the perfect matching polytopes of its tight cut contractions H_A and $H_{\overline{A}}$ we show how to decompose a vector in $\mathbb{Q}^{E(H)}$ into a pair of vectors in $\mathbb{Q}^{E(H_A)}$ and $\mathbb{Q}^{E(H_{\overline{A}})}$. Namely, every $e \in \delta_H(A)$ corresponds to a hyperedge $e_a \in E(H_A)$ and a hyperedge $e_{\overline{a}} \in E(H_{\overline{A}})$.

Thus, every $x \in \mathbb{Q}^{E(H)}$ corresponds to vectors $x^A \in \mathbb{Q}^{E(H_A)}$ and $x^{\overline{A}} \in \mathbb{Q}^{E(H_{\overline{A}})}$ via $x^A(e_a) = x^{\overline{A}}(e_{\overline{a}}) = x(e)$ for all $e \in \delta_H(A)$ and $x^A(e) = x(e)$, $x^{\overline{A}}(e') = x(e')$ for $e \subseteq \overline{A}$ and $e' \subseteq A$. On the other hand, if $y \in \mathbb{Q}^{E(H_A)}$ and $z \in \mathbb{Q}^{E(H_{\overline{A}})}$ are given with $y(e_a) = z(e_{\overline{a}})$ for all $e \in \delta_H(A)$, then we say that *y and z agree on $\delta_H(A)$*. In this case, we define a vector $y \oplus z \in \mathbb{Q}^{E(H)}$ by

$$(y \oplus z)(e) := \begin{cases} y(e) & \text{if } e \subseteq \overline{A} \\ z(e) & \text{if } e \subseteq A \\ y(e_a) & \text{if } e \in \delta_H(A) \end{cases}.$$

There is a one-to-one correspondence between vectors $x \in \mathbb{Q}^{E(H)}$ and pairs of vectors $y \in \mathbb{Q}^{E(H_A)}$, $z \in \mathbb{Q}^{E(H_{\overline{A}})}$ agreeing on $\delta_H(A)$.

Theorem 4.45. *If $\delta_H(A)$ is a tight cut in a matching covered hypergraph H and $x \in \mathbb{Q}^{E(H)}$, then x lies in the perfect matching polytope of H if and only if x^A lies in the perfect matching polytope of H_A and $x^{\overline{A}}$ in the perfect matching polytope of $H_{\overline{A}}$.*

Proof. First, suppose x lies in the perfect matching polytope of H. This means that there exist perfect matchings $M_1, \ldots, M_k$ and scalars $\lambda_1, \ldots, \lambda_k \geq 0$ with $x = \sum_{i=1}^k \lambda_i \chi^{M_i}$ and $\sum_{i=1}^k \lambda_i = 1$. Every perfect matching M_i corresponds to a pair of perfect matchings $M_{i,1}, M_{i,2}$ in H_A and $H_{\overline{A}}$ agreeing on $\delta_H(A)$. So the unique hyperedge $e \in M_i \cap \delta_H(A)$ corresponds to the hyperedges $e_a \in M_{i,1}$ and $e_{\overline{a}} \in M_{i,2}$. Furthermore, since $\delta_H(A)$ is a tight cut, it contains no other hyperedge of M_i. These observations imply that the vectors x^A and $x^{\overline{A}}$ can be written as $x^A = \sum_{i=1}^k \lambda_i \chi^{M_{i,1}}$ and $x^{\overline{A}} = \sum_{i=1}^k \lambda_i \chi^{M_{i,2}}$. Thus, x^A lies in the perfect matching polytope of H_A, and $x^{\overline{A}}$ in the perfect matching polytope of $H_{\overline{A}}$.

For the other direction, write x^A and $x^{\overline{A}}$ as a convex combination of characteristic vectors of perfect matchings: $x^A = \sum_{i=1}^k \lambda_i \chi^{M_i}$ and $x^{\overline{A}} = \sum_{i=1}^{k'} \lambda'_i \chi^{M'_i}$. We show that x lies in the perfect matching polytope of H by induction on $\max(k, k')$.

If $k = k' = 1$, then $x^A = \chi^{M_1}$, $x^{\overline{A}} = \chi^{M'_1}$, and M_1 and M'_1 agree on $\delta_H(A)$ as $x^A(e_a) = x^{\overline{A}}(e_{\overline{a}})$ for all $e \in \delta_H(A)$. This means that $x = \chi^M$ where M is the unique perfect matching in H corresponding to M_1, M'_1.

Now, suppose x lies in the perfect matching polytope of H if $\max(k, k') \leq t$.

For the induction step we assume that $k = \max(k, k') = t + 1$. Let $e^* \in \delta_H(A)$ be any hyperedge with $x^A(e^*_a) > 0$. We define sets $I_A := \{i \in [k] : e^*_a \in M_i\}$, and $I_{\overline{A}} := \{i \in [k'] : e^*_{\overline{a}} \in M'_i\}$. Because of $x^A(e^*_a) = x^{\overline{A}}(e^*_{\overline{a}})$, we know that $\sum_{i \in I_A} \lambda_i = \sum_{i \in I_{\overline{A}}} \lambda'_i$. We denote this value by Λ. If $\Lambda = 1$, then $I_A = [k]$, $I_{\overline{A}} = [k']$, and every pair of perfect matchings M_i and M'_j for $i \in [k]$, $j \in [k']$ agree on $\delta_H(A)$, which means that they correspond to a unique perfect matching in H. We denote this perfect matching by $M_{i,j}$. The following procedure writes x as a convex combination of perfect matchings in H:

While not all $\lambda_i = 0$ choose $i \in [k], j \in [k']$ with $\lambda_i > 0$ and $\lambda'_j > 0$, set $\mu_{i,j} := \min\{\lambda_i, \lambda'_j\}$, and decrease λ_i and λ'_j by $\mu_{i,j}$.

After every step of this construction $\sum_{i=1}^{k} \lambda_i$ and $\sum_{i=1}^{k'} \lambda'_k$ decrease by the same positive amount and at least one of λ_i or λ'_j becomes zero. Thus, the procedure will eventually stop with all λ_i and λ'_j equal to zero. Furthermore, we have $\sum_{i,j} \mu_{i,j}\chi^{M_{i,j}} = x$ and $\sum_{i,j} \mu_{i,j} = 1$.

If $\Lambda < 1$, then we look at the two vectors $y^A := \left(\sum_{i\in[k]\setminus I_A} \lambda_i \chi^{M_i}\right)/(1-\Lambda)$ and $y^{\overline{A}} = \left(\sum_{i\in[k']\setminus I_{\overline{A}}} \lambda'_i \chi^{M'_i}\right)/(1-\Lambda)$. The vector y^A lies in the perfect matching polytope of H_A, $y^{\overline{A}}$ lies in the perfect matching polytope of $H_{\overline{A}}$, and they are written as a convex combination of less than $t+1$ characteristic vectors of perfect matchings. Furthermore, for every $e \in \delta_H(A) \setminus \{e^*\}$ we have $y^A(e_a) = x^A(e_a) = x^{\overline{A}}(e_{\overline{a}}) = y^{\overline{A}}(e_{\overline{a}})$, and $y^A(e^*_a) = 0 = y^{\overline{A}}(e^*_{\overline{a}})$. By the induction hypothesis, the vector $y = y^A \oplus y^{\overline{A}}$ lies in the perfect matching polytope of H. On the other hand, also $z^A = \left(\sum_{i\in I_A} \lambda_i \chi^{M_i}\right)/\Lambda$, and $z^{\overline{A}} = \left(\sum_{i\in I_{\overline{A}}} \lambda'_i \chi^{M'_i}\right)/\Lambda$ define vectors of the perfect matching polytopes of H_A and $H_{\overline{A}}$ agreeing on $\delta_H(A)$ with less than $t+1$ summands in each convex combination. Thus, also $z = z^A \oplus z^{\overline{A}}$ lies in the perfect matching polytope of H. This implies that x is an element of the perfect matching polytope of H as $x = (1-\Lambda)y + \Lambda z$. □

There is no full description known of the perfect matching polytope of a general hypergraph in terms of linear equalities and inequalities. A relaxation is given by the *fractional perfect matching polytope* defined as

$$\mathcal{FP}_{\mathrm{PM}}(H) := \{x \in \mathbb{Q}^{E(H)} : x(\delta_H(v)) = 1 \ \forall v \in V(H), x \geq 0\}.$$

The perfect matching polytope of a hypergraph is a subset of its fractional perfect matching polytope, and both polytopes are equal if and only if the fractional perfect matching polytope is integral. This is for example the case for all hypergraphs generalizing bipartite graphs we considered in Chapter 1 except for partitioned hypergraphs. As a corollary of Theorem 4.45 we obtain that tight cut contractions preserve the property of having an integral fractional perfect matching polytope.

Corollary 4.46. *Let H be a matching covered hypergraph with a tight cut $\delta_H(A)$. If the fractional perfect matching polytope of H is integral, then the fractional perfect matching polytopes of the two contractions H_A and $H_{\overline{A}}$ are integral.*

Proof. We show that the fractional matching polytope of H_A is integral, the proof for $H_{\overline{A}}$ is symmetric. For an arbitrary vector $y \in \mathcal{FP}_{\mathrm{PM}}(H_A)$ we construct a vector

x lying in the perfect matching polytope of H such that $x(e) = y(e)$ for all $e \in E(H)$ with $e \subseteq \overline{A}$, and $x(e) = y(e_a)$ for all $e \in \delta_H(A)$. This implies that y lies in the perfect matching polytope of H_A by Theorem 4.45.

For every $e \in \delta_H(A)$ there exists a perfect matching M_e in $H_{\overline{A}}$ containing $e_{\overline{a}}$. The vector $z = \sum_{e\in\delta_H(A)} y(e_a)\chi^{M_e}$ lies in the perfect matching polytope of $H_{\overline{A}}$ because $\sum_{e\in\delta_H(A)} y(e_a) = y(\delta_{H_A}(a)) = 1$. Furthermore, y and z agree on $\delta_H(A)$ because $z(e_{\overline{a}}) = y(e_a)$ for every $e \in \delta_H(A)$. Thus, $x = y \oplus z$ satisfies all degree constraints for $v \in V(H)$. This means that it lies in the fractional perfect matching polytope of H and thus in its perfect matching polytope. We can write x as a convex combination of incidence vectors of perfect matchings $M_1, \ldots, M_k$:

$$x = \sum_{i=1}^{k} \lambda_i \chi^{M_i},\ \lambda_i \geq 0\ \forall i \in [k],\ \sum_{i=1}^{k} \lambda_i = 1.$$

If M'_i is the perfect matching of H_A corresponding to M_i, then $y = \sum_{i=1}^{k} \lambda_i \chi^{M'_i}$ holds, and thus y lies in the perfect matching polytope of H_A. □

In order to show that tight cut contractions of a uniformizable balanced hypergraph are balanced we give a characterization of balanced hypergraphs in terms of their fractional perfect matching polytopes. It seems that this characterization has not been published before though it follows easily from existing results on balanced hypergraphs.

Lemma 4.47. *A hypergraph H is balanced if and only if the fractional perfect matching polytope of each subhypergraph $H(S)$ for $S \subseteq V(H)$ is empty or integral.*

Proof. If H is balanced, then $H(S)$ is balanced for every $S \subseteq V(H)$, and its fractional matching polytope is integral, thus the matching polytope of H is given by

$$\{x \in \mathbb{Q}^{E(H(S))} : x(\delta_H(v)) \leq 1\ \forall v \in S, x \geq 0\},$$

see for example [Berge, 1984]. As the perfect matching polytope is a face of the matching polytope, the claim follows.

On the other hand, suppose H is not balanced. In this case H has a strong odd cycle. If S is the vertex set of such a cycle, then $H(S)$ contains an odd cycle (using hyperedges of size two) spanning S. The vector x defined by $x_e = 0.5$ for all hyperedges of size two of this odd cycle and $x_e = 0$ otherwise lies in the fractional perfect matching polytope of $H(S)$ but it cannot be written as a convex combination of incidence vectors of perfect matchings. □

Now, we prove that the tight cut contractions of balanced uniformizable hypergraphs are balanced.

Theorem 4.48. *Let H be a uniformizable balanced and matching covered hypergraph. If $A \subseteq V(H)$ is a set of vertices such that $\delta_H(A)$ is a tight cut, then the two contractions H_A and $H_{\overline{A}}$ are balanced.*

Proof. We show that H_A is balanced. The proof for $H_{\overline{A}}$ is symmetric.

Without loss of generality we assume that H is already uniform. If not, then there exists a function $m : V(H) \to \mathbb{Z}_{\geq 0}$ such that $H^{(m)}$ is uniform. The set $A^{(m)} := \{v^{(1)}, \ldots, v^{(m(v))} : v \in A\}$ defines a tight cut in $H^{(m)}$, and H_A is balanced if and only if $H^{(m)}_{A^{(m)}}$ is balanced.

By Lemma 4.47, the hypergraph H_A is balanced if the fractional perfect matching polytope of $H_A(S)$ is integral or empty for every $S \subseteq V(H_A)$. Let $S \subseteq V(H_A)$ such that the fractional perfect matching polytope of $H_A(S)$ is non-empty. If $a \notin S$, then $H_A(S) = H(S)$ and the claim follows as H is balanced. Thus, we assume $a \in S$ and set $T = (S \setminus \{a\}) \cup A$. Contracting A in the subhypergraph $H(T)$ of H gives a hypergraph that is isomorphic to $H_A(S)$. Let $x \in \mathbb{Q}^{E(H_A(S))}$ be a non-negative vector satisfying all degree constraints. For every $e \in \delta_H(A)$ let M_e be a perfect matching of H containing e. Every perfect matching M_e induces a perfect matching of $H(T)$ containing $e \cap T$. Set $N_e := \{f \in M_e : f \subseteq A\} \cup \{e \cap T\}$ and look at the vector $z \in \mathbb{Q}^{E(H(T))}$ defined by $z(f) = x(f)$ for $f \in E(H_A(S)) \cap E(H(T))$, and $z(e) = 0$ for all other $e \in E(H(T))$. Add $x(e_a \cap S)\chi^{N_e}$ to z for every $e \in \delta_H(A)$. Clearly, the resulting z is non-negative.

We show that z satisfies the degree constraints for all vertices $v \in T$. First, let $v \in S \setminus \{a\} = T \setminus A$. A hyperedge $e \in E(H(T))$ containing v lies either completely in $\overline{A}$ or it intersects both of $\overline{A}$ and A. In the first case, $e \in E(H_A(S))$ and $z(e) = x(e)$. In the second case, the hyperedge is of the form $e' \cap T$ for some $e' \in \delta_H(A)$ and $z(e' \cap T) = x(e'_a \cap S)$. In total we get that $z(\delta_{H(T)}(v)) = x(\delta_{H_A(S)}(v)) = 1$ for $v \in S \setminus \{a\}$. Now, consider a vertex $v \in A$. This vertex is incident to exactly one hyperedge in N_e for every $e \in \delta_H(A)$, and it is not incident to any hyperedge $f \in E(H_A(S)) \cap E(H(T))$. Thus,

$$z(\delta_{H(T)}(v)) = \sum_{e \in \delta_H(A)} x(e_a \cap S) = x(\delta_{H_A(S)}(a)) = 1.$$

As H is balanced, it follows that z lies in the perfect matching polytope of $H(T)$. Write z as a convex combination of incidence vectors of perfect matchings: $z = \sum_{i=1}^k \lambda_i \chi^{M_i}$, where M_i is a perfect matching of $H(T)$.

For every $i \in [k]$ we set

$$M'_i := \{f \in M_i \colon f \subseteq S \setminus \{a\}\} \cup \{e_a \cap S \colon e \in \delta_H(A), e \cap T \in M_i\}.$$

As $x(e) = z(e)$ for all $e \in E(H_A(S)) \cap E(H(T))$ and $x(e_a \cap S) = z(e \cap T)$ for all $e \in \delta_H(A)$, we get $x = \sum_{i=1}^k \lambda_i \chi^{M'_i}$. It remains to show that each M'_i is a perfect matching in $H_A(S)$. Every $v \in S \setminus \{a\}$ is covered exactly once by each M_i. Furthermore, the vertex a is contained in at least one of the hyperedges of M_i. Otherwise, no $e \in \delta_H(A)$ with $e \cap T \in M_i$ exists, implying that $e \subseteq A$ or $e \subseteq \overline{A}$ for $e \cap T \in M_i$, and the hyperedges $e \subseteq A$ of M_i form a matching covering A. But then $|A|$ is divisible by r, which is impossible as A defines a tight cut.

We show that a is covered by exactly one hyperedge of M_i. We get

$$1 = \sum_{e \in \delta_H(A)} x(e_a \cap S) = \sum_{i=1}^{k} \lambda_i \chi^{M_i}(\delta_{H_A}(a)) \geq \sum_{i=1}^{k} \lambda_i = 1.$$

It follows that $\chi^{M_i}(\delta_{H_A}(a)) = 1$ and M_i is a perfect matching in H_A. In total, we have shown that x can be written as a convex combination of incidence vectors of perfect matchings. □

We have seen that if $\delta_H(A)$ is a tight cut in a hypergraph H, then H is matching covered if and only if H_A and $H_{\overline{A}}$ are matching covered. In the graph case there is a larger class of cuts with this property, namely separating cuts. They can also defined for hypergraphs.

Definition 4.49. A cut $\delta_H(A)$ in a matching covered hypergraph H is called *separating* if H_A and $H_{\overline{A}}$ are matching covered.

Theorem 4.44 implies that every tight cut is separating. If a graph has a non-tight separating cut, then the non-negativity and degree constraints do not suffice to describe the perfect matching polytope or in other words the fractional perfect matching polytope of this graph is not integral.

If $\delta_H(A)$ is a separating cut in a hypergraph H, then the inequality $x(\delta_H(A)) \geq 1$ might not be valid for the perfect matching polytope. However, if H is uniformizable, then by similar arguments as in Lemma 4.42 we can show that $|M \cap \delta_H(A)| \geq 1$ holds for all perfect matchings M. Thus, $x(\delta_H(A)) \geq 1$ is a valid inequality for all vectors $x \in \mathcal{P}_{\mathrm{PM}}(H)$. As in the graph case, we use a separating cut that is not tight in a uniformizable hypergraph to construct a non-negative vector that satisfies all degree constraints but does not lie in the perfect matching polytope.

Theorem 4.50. *If H is a matching covered, uniformizable hypergraph with a separating cut $\delta_H(A)$ that is not tight, then the fractional perfect matching polytope is not integral.*

Proof. Suppose that H has a separating cut $\delta_H(A)$ that is not tight. We proceed as in the graph case, see [de Carvalho et al., 2004], by constructing a vector x lying in the fractional perfect matching polytope with $x(\delta_H(A)) < 1$.

Let M_0 be a perfect matching of H with $|M_0 \cap \delta_H(A)| \geq 2$. For every $e \in M_0$ let M_e be a perfect matching containing e and intersecting $\delta_H(A)$ in exactly one

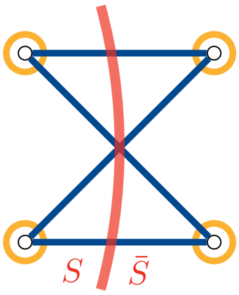


Figure 4.9: A unimodular hypergraph with a separating cut that is not tight.

hyperedge. These matchings exists because $\delta_H(A)$ is assumed to be a separating cut that is not tight. The vector $x \in \mathbb{Q}^{E(H)}$ defined by

$$x = \frac{1}{|M_0| - 1} \left(\sum_{e \in M_0} \chi^{M_e} - \chi^{M_0} \right)$$

is non-negative and satisfies $x(\delta_H(v)) = 1$ for all $v \in V(H)$ but

$$x(\delta_H(A)) = \frac{|M_0| - |M_0 \cap \delta_H(A)|}{|M_0| - 1} < 1.$$

Thus, $x \notin \mathcal{P}_{\mathrm{PM}}(H)$, which implies that the fractional perfect matching polytope of H is not integral. □

The previous theorem implies that uniformizable hypergraphs for which the non-negativity and degree constraints describe an integral polytope have no non-tight separating cuts. This holds in particular for unimodular, balanced, normal, and Mengerian hypergraphs.

Corollary 4.51. *If H is a uniformizable matching covered hypergraph with an integral fractional perfect matching polytope, then every separating cut of H is tight.*

If we consider hypergraphs that are not uniformizable, then it is possible that the fractional perfect matching polytope is integral but the hypergraph has a separating cut that is not tight. For example, if we take H to be the complete bipartite graph $K_{2,2}$ together with singleton hyperedges $\{v\}$ for every vertex v, then we obtain a unimodular hypergraph with a non-tight separating cut, see Figure 4.9.

Corollary 4.51 generalizes the fact that a bipartite graph has no non-tight separating cut, which can also be proven without polyhedral methods, see for example [de Carvalho et al., 2002]. Namely, if $\delta_G(A)$ is a non-tight separating cut in a graph G, then the subgraphs $G[A]$ and $G[\overline{A}]$ are non-bipartite. For hypergraphs we show that the shores of a non-tight separating cut cannot induce r-partite subhypergraphs.

Theorem 4.52. *Let H be a matching covered, r-uniform hypergraph and $A \subseteq V(H)$ be a set of vertices such that $\delta_H(A)$ is a non-tight separating cut. The subhypergraphs $H[A]$ and $H[\overline{A}]$ of H induced by A and $\overline{A}$ are not r-partite.*

Proof. We only show that $H[A]$ is not r-partite. The proof for $H[\overline{A}]$ is similar.

Suppose that there exists a partition $A_1 \cup A_2 \cup \ldots \cup A_r$ of A into r sets such that $|e \cap A_1| = \ldots = |e \cap A_r| = 1$ for all $e \in E(H[A])$. We choose any hyperedge $f \in \delta_H(A)$ and let M_f be a perfect matching of H with $M_f \cap \delta_H(A) = \{f\}$. The set $M'_f := \{e \in M_f : e \subseteq A\}$ is a matching of $H[A]$ covering $A \setminus f$. Without loss of generality we assume that there exists an index k with $1 \leq k \leq r-1$ such that f intersects $A_1, \ldots, A_k$ and f has an empty intersection with $A_{k+1}, \ldots, A_r$. It follows that $|A_1 \setminus f| = \ldots = |A_k \setminus f| = |A_{k+1}| = \ldots = |A_r|$. Furthermore, as we have chosen f arbitrarily we get that $|A_1 \setminus e| = \ldots = |A_k \setminus e| = |A_{k+1}| = \ldots = |A_r|$ for every $e \in \delta_H(A)$. Now, let M be a perfect matching of H intersecting $\delta_H(A)$ in more than one vertex. Again, the set $M' := \{e \in M : e \subseteq A\}$ forms a matching of $H[A]$. If $\{m_1, \ldots, m_s\} = M' \cap \delta_H(A)$, then M' covers the vertex set $A \setminus (m_1 \cup \ldots \cup m_s)$. This implies that $|A_1 \setminus (m_1 \cup \ldots \cup m_s)| = \ldots = |A_r \setminus (m_1 \cup \ldots \cup m_s)|$, which is impossible as $|A_1 \setminus (m_1 \cup \ldots \cup m_s)| = |A_1| - s \cdot |A_1 \cap f| < |A_r| = |A_r \setminus (m_1 \cup \ldots \cup m_s)|$. Thus, $H[A]$ is not r-partite. □

As a corollary, we directly obtain that an r-partite hypergraph cannot have a non-tight, separating cut.

Corollary 4.53. *If H is an r-partite matching covered hypergraph, then every separating cut is tight.*

Carvalho, Lucchesi, and Murty proved that also the reverse implication of Theorem 4.50 holds in the graph case. Namely, every graph with a non-integral fractional matching polytope contains a separating cut that is not tight. This is not true for hypergraphs of rank at least three. No 3-partite hypergraph has a non-tight, separating cut but there are 3-partite hypergraphs with a non-integral fractional perfect matching polytope, for example a complete 3-partite hypergraph.

4.4 Tight Cut Decomposition of Hypergraphs

In this section we generalize the tight cut decomposition from graphs to hypergraphs. We give an example of a hypergraph with two non-equivalent tight cut decompositions. However, for uniformizable hypergraphs we prove that the result of a tight cut decomposition is unique. Our proof goes along the lines of the one for graphs in [Lovász, 1987] using the fact that two crossing tight cuts can be "uncrossed".

Algorithm 1 Tight Cut Decomposition Procedure

```
1: procedure Tight Cut Decomposition($H$)
2:     $\mathcal{H} \leftarrow \{H\}$
3:     while $\exists H' \in \mathcal{H}$ s.t. $H'$ has a non-trivial tight cut do
4:         Choose any such $H'$ and $S \subseteq V(H')$ defining a non-trivial tight cut,
5:         $\mathcal{H} \leftarrow \mathcal{H} \setminus \{H'\} \cup \{H'_S, H'_{\overline{S}}\}$
6:     end while
7:     return $\mathcal{H}$
8: end procedure
```

First, we define formally what we mean by a tight cut decomposition by considering Algorithm 1. It takes a matching covered hypergraph as an input and decomposes it along a non-trivial tight cut if any exists. In this way, two new matching covered hypergraphs arise. If at least one of them has a non-trivial tight cut, then the algorithm contracts again both shores of this cut. In each execution of the while loop the number of hypergraphs increases by one. As we always choose a non-trivial tight cut in Step 4, the hypergraphs H'_S and $H'_{\overline{S}}$ have less vertices than H'. This implies that the algorithm will eventually terminate because a hypergraph with less than four vertices has no non-trivial tight cuts.

A *tight cut decomposition* of a matching covered hypergraph H is the output obtained by an execution of Algorithm 1. This means that a tight cut decomposition consists of a set of matching covered hypergraphs without non-trivial tight cuts that were obtained from H by successive tight cut contractions, see Figure 4.10.

We do not specify the concrete choice of the tight cut in Step 4 of Algorithm 1. Thus, different runs might give different outputs. We say that two tight cut decompositions $\mathcal{H}$ and $\mathcal{H}'$ are *equivalent* if there exists a bijection $\phi : \mathcal{H} \to \mathcal{H}'$ such that for every $H \in \mathcal{H}$ the hypergraphs H and $\phi(H)$ are isomorphic after deleting all but one copy of each set of parallel hyperedges. Any two tight cut decompositions of a graph are equivalent in this sense, which was shown in [Lovász, 1987]. This important result does not hold for general hypergraphs. A counterexample is depicted in Figure 4.11. In its center it shows a matching covered hypergraph H and two tight cuts $\delta_H(S)$, $\delta_H(T)$. The two hypergraphs obtained by contracting S and $\overline{S}$ are depicted to the left of H and the ones obtained by contracting T and $\overline{T}$ right of H. The resulting hypergraphs H_S, and $H_{\overline{S}}$, as well as, H_T, and $H_{\overline{T}}$ have only trivial tight cuts. Thus, $\{H_S, H_{\overline{S}}\}$ and $\{H_T, H_{\overline{T}}\}$ are two tight cut decompositions of H. As H_S is neither isomorphic to H_T, nor $H_{\overline{T}}$, the two tight cut decompositions are not equivalent.

However, we show that things work out well on uniformizable hypergraphs. Many proofs in the area of matching covered graphs depend on parity arguments utilizing the fact that every edge in a graph contains exactly two vertices. These arguments

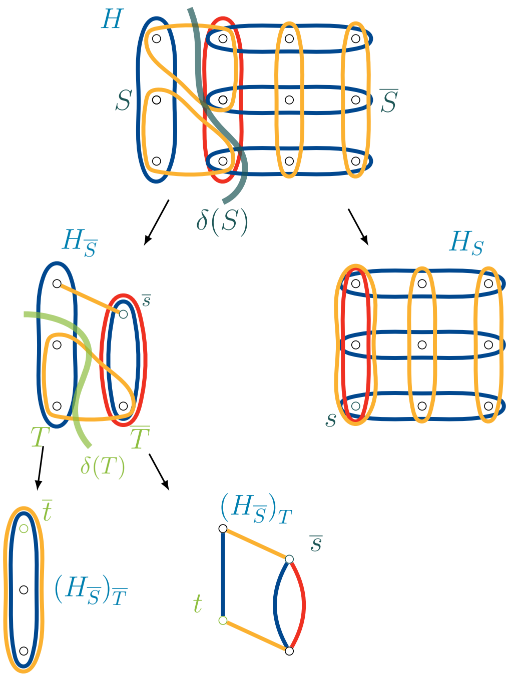


Figure 4.10: A Tight Cut Decomposition of a matching covered hypergraph.

can be generalized to uniformizable hypergraphs. In the remainder of this section we prove that uniformizable hypergraphs have a unique tight cut decomposition. Therefore, we describe tight cut decompositions in terms of families of non-trivial tight cuts with some additional properties. First, we show that a tight cut in a tight cut contraction corresponds to a tight cut in the original hypergraph.

Lemma 4.54. *Let H be a matching covered hypergraph with a non-trivial tight cut $\delta_H(S)$ defined by $S \subseteq V(H)$. A set $T \subseteq S$ defines a tight cut $\delta_H(T)$ in H if and only if it defines a tight cut $\delta_{H_{\overline{S}}}(T)$ in $H_{\overline{S}}$. Similar, $T \subseteq \overline{S}$ defines a tight cut in H if and only if it defines a tight cut in H_S.*

Proof. We only prove the claim for $T \subseteq S$ the other case is symmetric by interchanging S with $\overline{S}$.

The hypergraph $H_{\overline{S}}$ contains all $e \in E(H)$ with $e \subseteq S$ and for every $e \in \delta_H(S)$ it contains the hyperedge $e_{\overline{s}} = (e \cap S) \cup \{\overline{s}\}$, where $\overline{s}$ is a new vertex representing $\overline{S}$. For $T \subseteq S$ we get

$$\delta_{H_{\overline{S}}}(T) = \{e \in E(H) \colon e \subseteq S, e \cap T \neq \emptyset, e \setminus T \neq \emptyset\} \cup \{e_{\overline{s}} \colon e \in \delta_H(S), e \cap T \neq \emptyset\}.$$

Every perfect matching $M_{\overline{S}}$ of $H_{\overline{S}}$ corresponds to a unique perfect matching M of H such that $M_{\overline{S}} = \{e \in M \colon e \subseteq S\} \cup \{e^*_{\overline{s}}\}$, where e^* is the unique hyperedge in

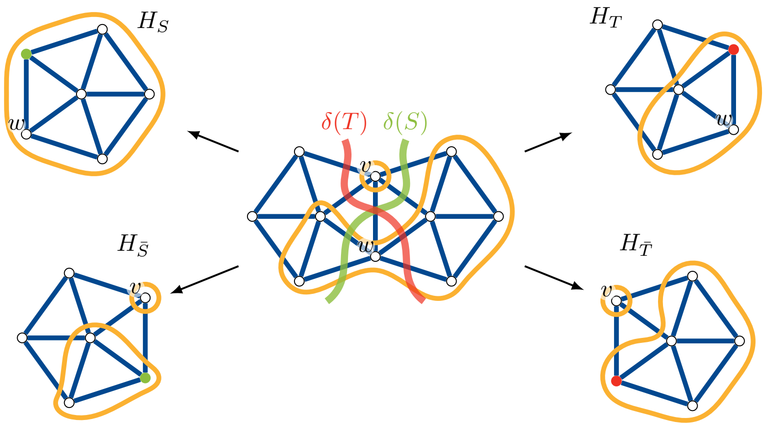


Figure 4.11: A non-uniform hypergraph with two non-equivalent tight cut decompositions.

$\delta_H(S) \cap M$, and vice versa. We show that $|\delta_{H_{\overline{S}}}(T) \cap M_{\overline{S}}| = |\delta_H(T) \cap M|$. Thus, $\delta_{H_{\overline{S}}}(T)$ is tight if and only if $\delta_H(T)$ is tight.

Every hyperedge $e \in \delta_H(T) \cap M$ has a non-empty intersection with S because $T \subseteq S$ and $T \cap e \neq \emptyset$. Thus, e is either a subset of S or it lies in the cut $\delta_H(S)$. In the first case, e is a hyperedge of $H_{\overline{S}}$ with $e \in \delta_{H_{\overline{S}}}(T) \cap M_{\overline{S}}$, and in the latter case $e_{\overline{s}} = e^*_{\overline{s}} \in \delta_{H_{\overline{S}}}(T) \cap M_{\overline{S}}$. Thus, $|\delta_{H_{\overline{S}}}(T) \cap M_{\overline{S}}| \geq |\delta_H(S) \cap T|$ holds.

On the other hand, if $e \in \delta_{H_{\overline{S}}}(T) \cap M_{\overline{S}}$, then either $e \subseteq S$ and $e \in \delta_H(T) \cap M$ or $e = e^*_{\overline{s}}$ and $e^* \in \delta_H(T)$. In total we get $|\delta_{H_{\overline{S}}}(T) \cap M_{\overline{S}}| = |\delta_H(T) \cap M|$. □

We say that two sets S, T *cross* if all four of $S \cap T, S \cap \overline{T}, \overline{S} \cap T, \overline{S} \cap \overline{T}$ are non-empty, otherwise S, T are called *non-crossing*.

Observe that a set S and its complement $\overline{S}$ define the same cut and the same tight cut contractions if $\delta_H(S)$ is tight. Thus, replacing S by $\overline{S}$ in Step 4 does not change the resulting tight cut decomposition. Now, two sets S and T are non-crossing if and only if one is contained in the other after possibly replacing one or both sets by their complements.

With the help of Lemma 4.54 we show that every tight cut decomposition of a hypergraph corresponds to a maximal family of pairwise non-crossing, non-trivial tight cuts.

Corollary 4.55. *Let H be a matching covered hypergraph. Every tight cut decomposition of H corresponds to a maximal family*

$$\mathcal{F} \subseteq \{S \subseteq V(H) \colon \delta_H(S) \textit{ is a non-trivial tight cut}\}$$

such that S and T are non-crossing and $S \neq \overline{T}$ for any two distinct sets $S, T \in \mathcal{F}$. On the other hand, every such family $\mathcal{F}$ defines a tight cut decomposition of H.

Proof. We prove the first claim by induction on $|\mathcal{H}|$.

If $|\mathcal{H}| = 1$, then H has no non-trivial tight cut, and $\mathcal{H} = \{H\}$ is its unique tight cut decomposition. This tight cut decomposition corresponds to $\mathcal{F} = \emptyset$, which is the unique maximal family of pairwise non-crossing, non-trivial tight cuts.

Suppose that the claim is true for every tight cut decomposition of size at most $k-1$ and let $\mathcal{H}$ be a tight cut decomposition of size k for some matching covered hypergraph H.

We associate to $\mathcal{H}$ a binary tree T in the following way. We start with a one vertex graph, which we call the root and we label it with H. At each execution of step 4 we add two new vertices labeled by H'_S and $H'_{\overline{S}}$ and connect them to the vertex representing H'. In this way, the leaves of T correspond to the elements of $\mathcal{H}$. Let $\delta_H(S)$ be the cut chosen in the first execution of the while loop. The unique path of a leaf of T to its root passes either through the vertex labeled by H_S or the one labeled by $H_{\overline{S}}$. Let $\mathcal{H}_1 \subseteq \mathcal{H}$ be the set of hypergraphs corresponding to leaves that fall into the first category and set $\mathcal{H}_2 := \mathcal{H} \setminus \mathcal{H}_1$.

The family $\mathcal{H}_1$ forms a tight cut decomposition of H_S and $\mathcal{H}_2$ a tight cut decomposition of $H_{\overline{S}}$. By the induction hypothesis, $\mathcal{H}_1$ corresponds to a maximal family $\mathcal{F}_1$ of pairwise non-crossing, non-trivial tight cuts of H_S, and $\mathcal{H}_2$ to a maximal family $\mathcal{F}_2$ of pairwise non-crossing, non-trivial tight cuts of $H_{\overline{S}}$. Lemma 4.54 implies that each $T \in \mathcal{F}_1 \cup \mathcal{F}_2$ defines a non-trivial tight cut in H. As $T \subseteq \overline{S}$ for all $T \in \mathcal{F}_1$, and $T \subseteq S$ for all $T \in \mathcal{F}_2$, the family $\mathcal{F} := \mathcal{F}_1 \cup \mathcal{F}_2 \cup \{S\}$ consists of pairwise non-crossing sets. If $\mathcal{F}$ is not maximal, there exists a non-trivial tight cut $\delta_H(T)$ such that $T \notin \mathcal{F}$ and T is non-crossing to all sets of $\mathcal{F}$. In particular, T is non-crossing to S, and thus $T \subseteq S$, $T \subseteq \overline{S}$, $\overline{T} \subseteq S$, or $\overline{T} \subseteq \overline{S}$. We can assume that $T \subseteq S$, the other cases are similar. In this case, T defines a non-trivial tight cut of $H_{\overline{S}}$ that is non-crossing to all elements of $\mathcal{F}_2$. This contradicts the maximality of $\mathcal{F}_2$. Thus, $\mathcal{F}$ is maximal.

For the second claim we use induction on $|\mathcal{F}|$.

If $|\mathcal{F}| = 0$, then H has no non-trivial tight cut, and thus $\{H\}$ is the unique tight cut decomposition of H.

Now, suppose the second claim is true if $|\mathcal{F}| \leq k-1$ and let $\mathcal{F}$ be a maximal family of pairwise non-crossing, non-trivial tight cuts of H of size k. Choose $S \in \mathcal{F}$ arbitrarily, and let H_S, $H_{\overline{S}}$ be the tight cut contractions of H with respect to $\delta_H(S)$. For every $T \in \mathcal{F}$ we have either $T \subseteq S$, $\overline{T} \subseteq S$, $T \subseteq \overline{S}$, or $\overline{T} \subseteq \overline{S}$. By Lemma 4.54, T or $\overline{T}$ define a tight cut of $H_{\overline{S}}$ in the first case and H_S in the second case. Set $\mathcal{F}_1 := \{T : T \in \mathcal{F}, T \subseteq S\} \cup \{\overline{T} : T \in \mathcal{F}, \overline{T} \subseteq S\}$, and $\mathcal{F}_2 := \{T : T \in \mathcal{F}, T \subseteq \overline{S}\} \cup \{\overline{T} : T \in \mathcal{F}, \overline{T} \subseteq \overline{S}\}$. Then, $\mathcal{F}_1$ is a family of pairwise non-crossing, non-trivial tight cuts of $H_{\overline{S}}$. If $\mathcal{F}_1$ is not maximal, then there exists a non-trivial tight cut $A \notin \mathcal{F}_1$ of $H_{\overline{S}}$ that does not cross any $T' \in \mathcal{F}_1$. But A does not cross any $T' \in \mathcal{F}_2$ and S, thus A and every element of $\mathcal{F}$ are non-crossing. This contradicts the maximality of $\mathcal{F}$. Thus, $\mathcal{F}_1$ is a maximal family of pairwise non-crossing, non-trivial tight cuts of $H_{\overline{S}}$. As $|\mathcal{F}_1| < |\mathcal{F}|$, the family $\mathcal{F}_1$ corresponds to a tight cut decomposition $\mathcal{H}_1$ of $H_{\overline{S}}$. By the same arguments, $\mathcal{F}_2$ corresponds to a tight cut decomposition $\mathcal{H}_2$ of H_S. Since we can choose S as the first cut in Algorithm 1, $\mathcal{H}_1 \cup \mathcal{H}_2$ is a tight cut decomposition of H corresponding to $\mathcal{F}$. □

An important property of tight cuts in graphs, which is the main ingredient of Lovász' uniqueness proof of the tight cut decomposition procedure, is the possibility to uncross two crossing tight cuts. That is, if S and T are crossing sets defining tight cuts in a graph G with $S \cap T$ of odd cardinality, then $\delta_G(S \cup T)$ and $\delta_G(S \cap T)$ are also tight. We show that a similar result holds for uniformizable hypergraphs.

Lemma 4.56. *Let H be a matching covered, uniformizable hypergraph, $S, T \subseteq V(H)$ be crossing sets such that $\delta_H(S)$ and $\delta_H(T)$ are tight. The cut $\delta_H(S \cap T)$ is tight if and only if $\delta_H(S \cup T)$ is tight.*

Proof. Suppose that $\delta_H(S \cap T)$ is tight but $\delta_H(S \cup T)$ is not tight. Furthermore, let $H^{(m)}$ be a multiplication of H that is r-uniform for some $r \in \mathbb{N}$. First, we show that there exists a perfect matching M with $|M \cap \delta_H(S \cup T)| \geq 2$. Let M' be a perfect matching with $|M' \cap \delta_H(S \cup T)| \neq 1$. If $|M' \cap \delta_H(S \cup T)| \geq 2$, then we choose $M = M'$. Otherwise, $M' \cap \delta_H(S \cup T) = \emptyset$. In this case, $\{e \subseteq S \cup T : e \in M'\}$ is a perfect matching of $H[S \cup T]$, thus $m(S \cup T) \equiv 0 \bmod r$. By Lemma 4.42, there exists a perfect matching M with $|M \cap \delta_H(S \cup T)| \geq 2$.

Every hyperedge $e \in \delta_H(S \cup T)$ lies either in exactly one of the two cuts $\delta_H(S)$, $\delta_H(T)$, or in both. Thus, $|M \cap \delta_H(S \cup T)| = 2$ and if e_1, e_2 are the two hyperedges in the intersection of M with $\delta_H(S \cup T)$, then we can assume that $e_1 \in \delta_H(S) \setminus \delta_H(T)$ and $e_2 \in \delta_H(T) \setminus \delta_H(S)$. It follows that $e_1 \cap S \neq \emptyset$, $e_1 \cap (\overline{S} \cap \overline{T}) \neq \emptyset$, $e_1 \subseteq T$ or $e_1 \subseteq \overline{T}$, thus $e_1 \subseteq \overline{T}$ and $e_1 \notin \delta_H(S \cap T)$. Similar, for e_2 we get $e_2 \cap T \neq \emptyset$, $e_2 \cap (\overline{S} \cap \overline{T}) \neq \emptyset$, $e_2 \subseteq \overline{S}$, and thus $e_2 \notin \delta_H(S \cap T)$. This is a contradiction to the tightness of $\delta_H(S \cap T)$ because $M \cap \delta_H(S \cap T) \subseteq (M \cap \delta_H(S)) \cup (M \cap \delta_H(T)) = \{e_1, e_2\}$, and therefore $M \cap \delta_H(S \cap T) = \emptyset$.

The other direction follows from $\delta_H(\overline{S}) = \delta_H(S)$, $\delta_H(\overline{T}) = \delta_H(T)$, $\delta_H(\overline{S} \cap \overline{T}) = \delta_H(S \cup T)$, and $\delta_H(\overline{S} \cup \overline{T}) = \delta_H(S \cap T)$. □

If we apply this lemma to S and $\overline{T}$ and observe that $\overline{S \cup \overline{T}} = \overline{S} \cap T$ we get the following corollary.

Corollary 4.57. *Let H be a matching covered uniformizable hypergraph, and $S, T \subseteq V(H)$ be crossing sets such that $\delta_H(S)$ and $\delta_H(T)$ are tight. The cut $\delta_H(S \cap \overline{T})$ is tight if and only if $\delta_H(\overline{S} \cap T)$ is tight.*

Lemma 4.58. *Let H be a matching covered uniformizable hypergraph, and $S, T \subseteq V(H)$ be crossing sets such that $\delta_H(S)$ and $\delta_H(T)$ are tight. If $\delta_H(S \cap T)$ is not tight, then $\delta_H(S \cap \overline{T})$ is tight.*

Proof. If $\delta_H(S \cap T)$ is not tight, then also $\delta_H(S \cup T)$ is not tight. Using the same arguments as in the proof of Lemma 4.56 we can find a perfect matching M with $|M \cap \delta_H(S \cup T)| = 2$. We have seen that such a matching does not intersect $\delta_H(S \cap T)$. This implies that $m(S \cap T) \equiv 0 \mod r$ where $H^{(m)}$ is an r-uniform multiplication of H.

Suppose that $\delta_H(S \cap \overline{T})$ is not tight. By a symmetric argument, we find a perfect matching M' with $|M' \cap \delta_H(S \cap \overline{T})| = 0$, which implies that $m(S \cap \overline{T}) \equiv 0 \mod r$. Now, $m(S) = m(S \cap \overline{T}) + m(S \cap T) \equiv 0 \mod r$ contradicting Lemma 4.42. □

We sum up the previous results in the following corollary.

Corollary 4.59. *If H is a matching covered uniformizable hypergraph, and $S, T \subseteq V(H)$ are crossing sets such that $\delta_H(S)$ and $\delta_H(T)$ are tight, then $\delta_H(S \cap T)$ and $\delta_H(S \cup T)$, or $\delta_H(S \cap \overline{T})$ and $\delta_H(\overline{S} \cap T)$ are tight.*

If S, T are crossing sets defining tight cuts in a uniformizable hypergraph, then we can always assume that $S \cap T$ and $S \cup T$ define tight cuts after possibly replacing T by $\overline{T}$. In the graph case, all four sets $S, T, S \cap T$, and $S \cup T$ would be of odd size. If $\delta_H(S) \neq \delta_H(T)$, then we show a similar result for uniformizable hypergraphs. Therefore, we need the following observation concerning tight cuts.

Observation 4.60. *If S defines a tight cut in a uniformizable matching covered hypergraph H and $\delta_H(A)$ is a non-empty cut with $\delta_H(A) \subseteq \delta_H(S)$, then $\delta_H(A) = \delta_H(S)$.*

Proof. Suppose there exists a hyperedge $e^* \in \delta_H(S) \setminus \delta_H(A)$. Let M be a perfect matching of H containing e^*. As $|M \cap \delta_H(S)| = 1$, it follows that $M \cap \delta_H(A) = \emptyset$. This implies that $m(A) \equiv 0 \mod r$ for any function $m : V(H) \to \mathbb{N}$ and $r \in \mathbb{N}$ such that $H^{(m)}$ is r-uniform. By Lemma 4.42, there exists a perfect matching M' with $|M' \cap \delta_H(A)| \geq 2$. Thus, $|M' \cap \delta_H(S)| \geq 2$, contradicting that $\delta_H(S)$ is tight. □

Lemma 4.61. *Let H be a matching covered, uniformizable hypergraph together with a function $m : V(H) \to \mathbb{N}$ such that $H^{(m)}$ is r-uniform for some $r \in \mathbb{Z}$, and $S, T \subseteq V(H)$ be two crossing sets such that S, T and $S \cup T$ define tight cuts in H. If $\delta_H(S) \neq \delta_H(T)$, then $m(S) \equiv m(S \cap T) \equiv m(S \cup T) \equiv m(T) \mod r$.*

Proof. We first show that $\delta_H(S) \neq \delta_H(T)$ implies $\delta_H(S) \neq \delta_H(S \cap T)$ and $\delta_H(S) \neq \delta_H(S \cup T)$. Suppose that $\delta_H(S) = \delta_H(S \cap T)$. As S and T are crossing, the set $S \cap \overline{T}$ is non-empty and because H is connected $\delta_H(S \cap \overline{T}) \neq \emptyset$. Let $e \in \delta_H(S \cap \overline{T})$. If $e \notin \delta_H(S)$, then $e \subseteq S$ or $e \subseteq \overline{S}$. Because of $e \in \delta_H(S \cap \overline{T})$ we have $e \subseteq S$ and

$e \cap (S \cap T) \neq \emptyset$. But then $e \in \delta_H(S \cap T) \setminus \delta_H(S)$. Thus, $\delta_H(S \cap \overline{T}) \subseteq \delta_H(S)$, and by Observation 4.60 equality holds. Now, $e \in \delta_H(S)$ implies $e \in \delta_H(S \cap T) = \delta_H(S \cap \overline{T})$, and thus $e \cap (S \cap T) \neq \emptyset$ and $e \cap (S \cap \overline{T}) \neq \emptyset$, implying that $e \in \delta_H(T)$. Together with Observation 4.60 it follows that $\delta_H(S)$ and $\delta_H(T)$ coincide, a contradiction.

If we replace S and T by their complements, then the argument above shows that $\delta_H(S) = \delta_H(\overline{S}) \neq \delta_H(\overline{S} \cap \overline{T}) = \delta_H(S \cup T)$.

Now, let $e_1 \in \delta_H(S) \setminus \delta_H(S \cap T)$, and choose a perfect matching M containing e_1. Let $e_2 \in M \cap \delta_H(S \cap T)$. It follows that $e_2 \in \delta_H(T)$, $e_2 \notin \delta_H(S)$, $e_2 \notin \delta_H(S \cup T)$, $e_1 \notin \delta_H(T)$, $e_1 \in \delta_H(S \cup T)$. Thus, $e_1 \cap (\overline{S} \cap \overline{T}) \neq \emptyset$ and $e_1 \subseteq \overline{T}$, which implies

$$\sum_{v \in S} m(v) \equiv \sum_{v \in e_1 \cap S} m(v) = \sum_{v \in e_1 \cap (S \cup T)} m(v) \equiv \sum_{v \in S \cup T} m(v) \bmod r.$$

For e_2 we get $e_2 \cap (S \cap T) \neq \emptyset$ and $e_2 \subseteq S$, thus

$$\sum_{v \in T} m(v) \equiv \sum_{v \in e_2 \cap T} m(v) = \sum_{v \in e_2 \cap (S \cap T)} m(v) \equiv \sum_{v \in S \cap T} m(v) \bmod r.$$

Starting the same argument with $e_1 \in \delta_H(S) \setminus \delta_H(S \cup T)$ gives $m(S) \equiv m(S \cap T) \bmod r$ and $m(T) \equiv m(S \cup T) \bmod r$. □

By Corollary 4.55, every tight cut decomposition is determined by a maximal family of pairwise non-crossing, non-trivial tight cuts. We show that each of these families gives rise to the same list of hypergraphs in a tight cut decomposition. The proof works along the lines of the one for graphs by Lovász. Its main ingredient is Corollary 4.59, which implies that for two crossing cuts $\delta_H(S)$, $\delta_H(T)$ we can find two cuts $\delta_H(U_1)$, $\delta_H(U_2)$ such that U_1, S and T, as well as U_2, S and T are pairwise non-crossing. If one of U_1, U_2 is non-trivial, then we use induction. A problem occurs if U_1 and U_2 are both trivial. We can assume without loss of generality that $U_1 = S \cap T$ and $U_2 = S \cup T$. In this case, U_1 and U_2 are trivial if and only if $|S \cap T| = 1$ and $|V(H) \setminus (S \cup T)| = 1$. Figure 4.12 shows an example for this case, where also the tight cut contractions with respect to $\delta_H(S)$ and $\delta_H(T)$ are drawn. In this example $H_{\overline{S}}$ is isomorphic to H_T, and H_S is isomorphic to $H_{\overline{T}}$. In the following lemma we prove that this is always the case.

Lemma 4.62. *Let H be a matching covered, uniformizable hypergraph, $\delta_H(S), \delta_H(T)$ be crossing tight cuts with $\delta_H(S) \neq \delta_H(T)$. If $\delta_H(S \cap T)$ and $\delta_H(S \cup T)$ are trivial tight cuts, then H_S and $H_{\overline{T}}$ as well as $H_{\overline{S}}$ and H_T are isomorphic up to parallel hyperedges.*

Proof. First, we show that if $e \in \delta_H(S) \cap \delta_H(T)$, then $e = (S \cap T) \cup (\overline{S} \cap \overline{T})$. As H is uniformizable there exists a function $m : V(H) \to \mathbb{N}$ such that $H^{(m)}$ is r-uniform for some $r \in \mathbb{Z}$. Let v^* be the unique vertex in $S \cap T$ and w^* be the unique vertex

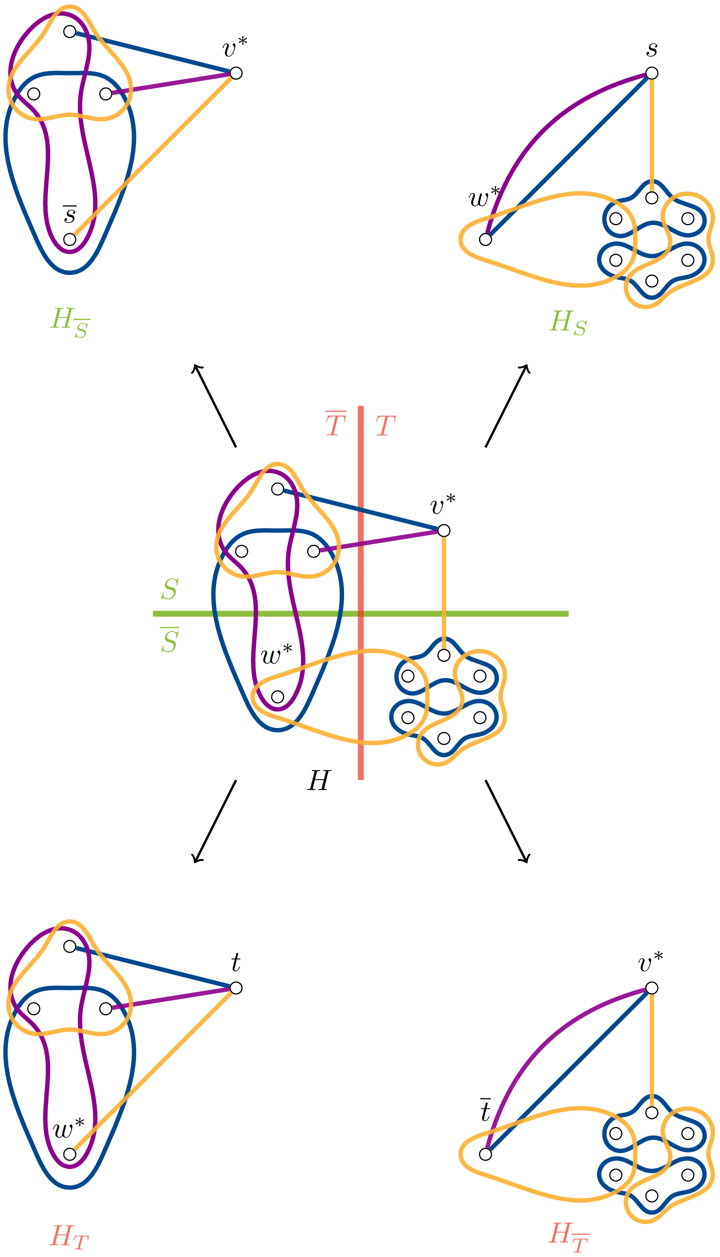


Figure 4.12: Two tight cuts $\delta_H(S)$, $\delta_H(T)$ such that $\delta_H(S \cap T)$ and $\delta_H(S \cup T)$ are trivial cuts. The tight cut contractions with respect to $\delta_H(S)$ are drawn above H, and the tight cut contractions with respect to $\delta_H(T)$ below H.

in $\overline{S} \cap \overline{T}$. If $e \in \delta_H(S) \cap \delta_H(T)$, then $e \in \delta_H(S \cap T) \cap \delta_H(S \cup T)$. Thus, $v^* \in e$ and $w^* \in e$. By Lemma 4.61, $m(v^*) = m(S) = m(T)$ and $m(w^*) = r - m(v^*)$, and therefore $e = \{v^*, w^*\}$.

Now, we show that H_S and $H_{\overline{T}}$ are equivalent up to multiple hyperedges. We define a function $\phi : V(H_{\overline{T}}) \to V(H_S)$ by

$$\phi(v) := \begin{cases} v, & v \in \overline{S} \cap T \\ s, & v = v^* \in S \cap T \\ w^*, & v = \overline{t} \in V(H_{\overline{T}}) \setminus T \end{cases}.$$

Observe, that $V(H_{\overline{T}}) = (\overline{S} \cap T) \cup \{v^*\} \cup \{\overline{t}\}$ and $V(H_S) = (\overline{S} \cap T) \cup \{w^*\} \cup \{s\}$. Thus, ϕ is well-defined and bijective. We claim that ϕ extends to a function from $E(H_{\overline{T}})$ to $E(H_S)$ via $\phi(e) = \{\phi(v) : v \in e\}$ for all $e \in E(H_{\overline{T}})$. First, we show that ϕ is well-defined.

- If $e \subseteq \overline{S} \cap T$, then $\phi(e) = e \in E(H_S)$.

- If $e \subseteq T$ and $v^* \in e$, then $\phi(e) = \left(e \cap \overline{S}\right) \cup \{s\} \in E(H_S)$.

- If $\overline{t} \in e$, then there exists a hyperedge $\tilde{e} \in E(H)$ with $(\tilde{e} \cap T) \cup \{\overline{t}\} = e$. If $\tilde{e} = \{v^*, w^*\}$, then $e = \{v^*, \overline{t}\}$, and $\phi(e) = \{s, w^*\} = (\tilde{e} \cap \overline{S}) \cup \{s\} \in E(H_S)$. Next, we consider the case $\tilde{e} \cap \{v^*, w^*\} = \{w^*\}$. In this case, $\tilde{e} \in \delta_H(T) \setminus \delta_H(S)$ and thus $\tilde{e} \subseteq \overline{S}$. We get $\phi(e) = (\tilde{e} \cap \overline{S} \cap T) \cup \{w^*\} = e \setminus \{t\} \cup \{w^*\} = \tilde{e} \in E(H_S)$. It remains to consider the case $\tilde{e} \cap \{v^*, w^*\} = \{v^*\}$. Now, $\tilde{e} \in \delta_H(T) \setminus \delta_H(S)$ and $v^* \in \tilde{e}$ implies $e \subseteq S$. Let $\widetilde{M}$ be a perfect matching containing $\tilde{e}$ and $e' \in \widetilde{M} \cap \delta_H(S)$. It follows that $w^* \in e'$, and $e' \in \delta_H(S) \setminus \delta_H(T)$ thus $e' \subseteq \overline{T}$. Furthermore, $\left(e' \cap \overline{S}\right) \cup \{s\} = \{w^*, s\}$ implies $\{w^*, s\} \in E(H_S)$. Now, $\phi(e) = (\tilde{e} \cap \overline{S} \cap T) \cup \{s\} \cup \{w^*\} = \{s, w^*\} \in E(H_S)$.

Next, we show that $\phi : E(H_{\overline{T}}) \to E(H_S)$ is surjective, i.e., for every $e \in E(H_S)$ there exists $f \in E(H_{\overline{T}})$ such that $\phi(f) = e$.

- If $e \subseteq \overline{S} \cap T$, then $e \in E(H_{\overline{T}})$ and $\phi(e) = e$.

- If $e \subseteq \overline{S}$ and $w^* \in e$, then $e \in \delta_H(T)$, thus $(e \cap T) \cup \{\overline{t}\} \in E(H_{\overline{T}})$, and $\phi((e \cap T) \cup \{\overline{t}\}) = e$.

- If $s \in e, w^* \notin e$, then $e = \tilde{e} \cap \overline{S} \cup \{s\}$ where $\tilde{e} \in \delta_H(S)$ with $v^* \in \tilde{e}$ and $w^* \notin \tilde{e}$. It follows that $\tilde{e} \notin \delta_H(T)$ and as $v^* \in \tilde{e}$ this means $\tilde{e} \subseteq T$. Thus, $\tilde{e} \in E(H_{\overline{T}})$ and $\phi(\tilde{e}) = \left(\tilde{e} \cap \overline{S} \cap T\right) \cup \{s\} = \left(\tilde{e} \cap \overline{S}\right) \cup \{s\} = e$.

- If $s, w^* \in e$, then $e = \left(\tilde{e} \cap \overline{S}\right) \cup \{s\}$ where $\tilde{e} \in \delta_H(S)$ with $w^* \in \tilde{e}$. If also $v^* \in \tilde{e}$, then $\tilde{e} = \{v^*, w^*\}$ and $\{v^*, \bar{t}\} \in E(H_{\overline{T}})$ with $\phi(\{v^*, \bar{t}\}) = \{s, w^*\} = e$.

  If $v^* \notin \tilde{e}$, we know that $\tilde{e} \notin \delta_H(T)$ and as $w^* \in \tilde{e}$ we get $\tilde{e} \subseteq \overline{T}$. This means that $e = \{s, w^*\}$. Now, let $\widetilde{M}$ be a perfect matching containing $\tilde{e}$ and $\tilde{f} \in M \cap \delta_H(T)$. We get $\tilde{f} \subseteq S$ because $v^* \in \tilde{f}$ and $\tilde{f} \notin \delta_H(S)$. Therefore, $\{v^*, t\} = (\tilde{f} \cap T) \cup \{t\} \in E(H_{\overline{S}})$ and $\phi(\{v^*, t\}) = \{s, w^*\} = e$.

It remains to show that $\phi(f) = \phi(f')$ implies that f and f' are parallel. This is clear if $\phi(f) \subseteq \overline{S} \cap T$ as ϕ is the identity on $\overline{S} \cap T$. If $\phi(f) \subseteq \overline{S}$ and $w^* \in \phi(f)$, then $\bar{t} \in f \cap f'$ and $f \cap T = f' \cap T$, i.e., f and f' are parallel. Now, if $s \in \phi(f)$ and $w^* \notin \phi(f)$, then $v^* \in f \cap f'$, and $\bar{t} \notin f$, $\bar{t} \notin f'$. This implies that $f, f' \subseteq T$ and thus $f \setminus \{v^*\} = f' \setminus \{v^*\}$. In total, f and f' contain the same vertices and are therefore parallel. If both s and w^* lie in $\phi(f)$, then $\phi(f) = \{s, w^*\}$ and $f = \{v^*, \bar{t}\} = f'$.

In total, we have shown that $H_{\overline{T}}$ and H_S are isomorphic via ϕ up to parallel hyperedges.

By similar arguments, we can show that H_T and $H_{\overline{S}}$ are isomorphic up to parallel hyperedges using the function $\phi : V(H_T) \to V(H_{\overline{S}})$ defined by

$$\phi(v) := \begin{cases} v, & v \in S \cap \overline{T} \\ \bar{s}, & v = w^* \in \overline{S} \cap \overline{T} \\ v^*, & v = t \in V(H_T) \setminus \overline{T} \end{cases} .$$

□

Using the previous lemmata and corollaries we prove our main result; the uniqueness of the tight cut decomposition in matching covered, uniformizable hypergraphs.

Theorem 4.63. *Any two tight cut decomposition procedures of a matching covered, uniformizable hypergraph yield the same list of indecomposable hypergraphs up to multiplicity of parallel hyperedges.*

Proof. As in the graph case, we use induction on the number of vertices of H. If H has at most three vertices, then H has no non-trivial tight cut. Now, suppose the theorem holds for all hypergraphs H with $|V(H)| \leq l$. Let H be a hypergraph on $(l+1)$ vertices and $\mathcal{F}, \mathcal{F}'$ be two maximal families of pairwise non-crossing, non-trivial tight cuts. If H has no non-trivial tight cut, then $\mathcal{F} = \mathcal{F}' = \emptyset$. Otherwise, we distinguish the following cases where the first and second one are identical to the ones in the graph case.

1. $\mathcal{F}$ and $\mathcal{F}'$ have a common member S. We can start a tight cut decomposition procedure with S resulting in the matching covered, uniformizable hypergraphs

H_S and $H_{\overline{S}}$ with at most l vertices. By induction hypothesis, the families

$$\mathcal{F}_1 := \{T : T \in \mathcal{F}, T \subseteq S\} \cup \{\overline{T} : T \in \mathcal{F}, \overline{T} \subseteq S\},$$
$$\mathcal{F}'_1 := \{T : T \in \mathcal{F}', T \subseteq S\} \cup \{\overline{T} : T \in \mathcal{F}', \overline{T} \subseteq S\}$$

induce equivalent tight cut decompositions of $H_{\overline{S}}$, and

$$\mathcal{F}_2 := \{T : T \in \mathcal{F}, T \subseteq \overline{S}\} \cup \{\overline{T} : T \in \mathcal{F}, \overline{T} \subseteq \overline{S}\},$$
$$\mathcal{F}'_2 := \{T : T \in \mathcal{F}', T \subseteq \overline{S}\} \cup \{\overline{T} : T \in \mathcal{F}', \overline{T} \subseteq \overline{S}\}$$

induce equivalent tight cut decompositions of H_S. Thus, the decomposition procedures associated to $\mathcal{F}$ and $\mathcal{F}'$ yield the same list of indecomposable hypergraphs.

2. There exist $S \in \mathcal{F}$, $T \in \mathcal{F}'$ such that S and T are non-crossing. Let $\mathcal{F}''$ be any maximal family of pairwise non-crossing, non-trivial tight cuts containing both S and T. By the first case, every tight cut decomposition associated to $\mathcal{F}$ and $\mathcal{F}''$, as well as $\mathcal{F}'$ and $\mathcal{F}''$ yield the same list of indecomposable hypergraphs.

3. Suppose that there exists $S \in \mathcal{F}$, $T \in \mathcal{F}'$ such that S and T are crossing and $\delta_H(S) = \delta_H(T)$. Then, $\delta_H(S \cap T) = \delta_H(S \cap \overline{T}) = \delta_H(\overline{S} \cap \overline{T}) = \delta_H(\overline{S} \cap T) = \delta_H(S)$. If one of the four sets $S \cap T, S \cap \overline{T}, \overline{S} \cap T, \overline{S} \cap \overline{T}$ has size at least two, then it defines a non-trivial tight cut non-crossing to S and T. We consider any maximal family of pairwise non-crossing, non-trivial tight cuts containing this cut. By the first case, $\mathcal{F}''$ and $\mathcal{F}$, as well as $\mathcal{F}''$ and $\mathcal{F}'$ induce equivalent tight cut decompositions.

   If all four of the sets $S \cap T, S \cap \overline{T}, \overline{S} \cap T, \overline{S} \cap \overline{T}$ have size one, then $|S| = |T| = 2$, $|V(H)| = 4$, and H consists of hyperedges of the form $e = V(H)$. In this case, each of $H_S, H_{\overline{S}}, H_T, H_{\overline{T}}$ has three vertices and some parallel hyperedges containing all three vertices. As a hypergraph with at most three vertices has only trivial cuts, we have $\mathcal{F} = \{S\}$ and $\mathcal{F}' = \{T\}$ and the tight cut contractions with respect to $\delta_H(S)$ and $\delta_H(T)$ are isomorphic.

4. Now, we consider the case $\delta_H(S) \neq \delta_H(T)$, and S, T are crossing sets for all $S \in \mathcal{F}$, $T \in \mathcal{F}'$. By Corollary 4.59, we can assume $S \cap T$ and $S \cup T$ define tight cuts. If $|S \cap T| > 1$ or $|V(H) \setminus (S \cup T)| > 1$, then $U = S \cap T$ or $U = S \cup T$ defines a non-trivial tight cut that neither crosses S nor T. Let $\mathcal{F}''$ be a maximal family of pairwise non-crossing, non-trivial tight cuts containing U. By the second case, $\mathcal{F}''$ and $\mathcal{F}$, as well as $\mathcal{F}''$ and $\mathcal{F}'$ yield equivalent tight cut decompositions.

   Next, we assume that $|S \cap T| = 1$ and $|\overline{S} \cap \overline{T}| = 1$. If $\mathcal{F}$ and $\mathcal{F}'$ consist just of one cut, then by Lemma 4.62 the tight cut contractions with respect to $\delta_H(S)$ and $\delta_H(T)$ yield isomorphic hypergraphs. Otherwise, we may assume by symmetry that $\mathcal{F}$ contains another tight cut $S' \notin \{S, \overline{S}\}$. As S' and S are non-crossing, we can assume $S' \cap S = \emptyset$, i.e., $S' \subseteq \overline{S}$. If S' and T are non-crossing, then we are in the second case. If both sets are crossing but $\delta_H(S') = \delta_H(T)$, then we are in the third case.

In the remainder of the proof, we assume that S' and T are crossing sets defining different cuts. We know that $S' \cap T \neq \emptyset$ and $S' \cap \overline{T} \neq \emptyset$. It follows that $S' \cap \overline{T} = S' \cap \overline{S} \cap \overline{T} = \overline{S} \cap \overline{T}$ because $|\overline{S} \cap \overline{T}| = 1$. Let $w^* \in \overline{S} \cap \overline{T}$. We can write S' as $S' = \{w^*\} \cup (S' \cap T)$. Suppose that $\delta_H(S' \cap T)$ is a tight cut, and let $m : V(H) \to \mathbb{N}$ be a function such that $H^{(m)}$ is r-uniform for some $r \in \mathbb{N}$. By Lemma 4.61 and the assumption $\delta_H(S') \neq \delta_H(T)$, we have $m(S') \equiv m(S' \cap T) \equiv m(T) \equiv m(S \cap T) \bmod r$. On the other hand,

$$\begin{aligned} m(S') &= m(S' \cap T) + m(S' \cap \overline{T}) = m(S' \cap T) + m(\overline{S} \cap \overline{T}) \\ &\equiv m(T) - m(S \cap T) \equiv 0 \bmod r. \end{aligned}$$

However, as $\delta_H(S')$ is a tight cut $m(S')$ cannot be divisible by r. Thus, $\delta_H(S' \cap T)$ is not tight and therefore $\delta_H(\overline{S}' \cap T)$ is a tight cut by Corollary 4.59. This cut is non-trivial because otherwise $S' \cap T = \overline{S} \cap T$ would follows, which implies that $S' = \{w^*\} \cup (S' \cap T) = \overline{S}$, contradicting the choice of S'. This means that $U = \overline{S}' \cap T$ defines a non-trivial tight cut that does not cross any cut of $\mathcal{F}$ or $\mathcal{F}'$. Again, by considering a maximal family $\mathcal{F}''$ of pairwise non-trivial tight cuts containing U and using the second case, we get that $\mathcal{F}$ and $\mathcal{F}'$ yield equivalent tight cut decompositions. □

4.5 Complexity Results

Though it is $\mathcal{NP}$-hard to find a perfect matching of maximum weight in a hypergraph, it can be solved quite efficiently in practice within a branch-and-cut framework in an integer programming solver. However, the performance heavily depends on the size of the input hypergraph. Therefore, it is of great use if one can decompose the perfect matching problem into smaller ones. By Theorem 4.45, every tight cut yields such a decomposition. In graphs one can find a tight cut decomposition in polynomial time as described for example in Section 2 of [Edmonds et al., 1982]. For hypergraphs it is not clear how to generalize this result even in the uniform case. In the case of balanced uniformizable hypergraphs, we give a polynomial time algorithm to find a non-trivial tight cut based on submodular function minimization.

First, we observe that the problem to decide whether a cut is tight in a uniformizable hypergraph can be reduced to the uniform case. Let H be a uniformizable hypergraph, and $S \subseteq V(H)$. The cut $\delta_H(S)$ defined by S is tight if and only if the set $S^{(m)} := \{v^{(1)}, \ldots, v^{(m(v))} : v \in S\}$ defines a tight cut in $H^{(m)}$, where $H^{(m)}$ is a uniform multiplication of H. We assume that H is uniform in the remainder of this section and give a polynomial time algorithm that finds a non-trivial tight cut in a uniform, balanced, matching covered hypergraph.

In such a hypergraph every separating cut, which is a cut $\delta_H(S)$ such that H_S and $H_{\overline{S}}$ are matching covered, is tight by Corollary 4.51. We can exploit this fact to compute tight cuts on uniform balanced hypergraphs as follows:

For every $e \in E(H)$ we compute a perfect matching M_e containing e. This can be done in polynomial time by linear programming as the fractional perfect matching polytope of a balanced hypergraph is integral. Now, if $\delta_H(S)$ is a tight cut, then $|M_e \cap \delta_H(S)| = 1$ for all $e \in E(H)$. On the other hand, if $\delta_H(S)$ is a cut such that $|M_e \cap \delta_H(S)| = 1$ for all $e \in E(H)$, then $\delta_H(S)$ is a separating cut and also a tight cut as H is a uniform balanced hypergraph. In total, $\delta_H(S)$ is tight if and only if $|M_e \cap \delta_H(S)| = 1$ for all $e \in E(H)$, where $\{M_e : e \in E(H)\}$ is an arbitrary set of perfect matchings such that M_e contains e for every $e \in E(H)$.

For every $e \in E(H)$ we choose a perfect matching M_e such that $e \in M_e$, and define a weight function $w : E(H) \to \mathbb{Z}$ by $w(f) := |\{e \in E(H) : f \in M_e\}|$ for every $f \in E(H)$. The value $w(f)$ of the function w at a hyperedge f is equal to the number of perfect matchings in the set $\{M_e : e \in E(H)\}$ containing f. If $S \subseteq V(H)$ is such that its size is not divisible by r, then $|M \cap \delta_H(S)| \geq 1$ for all perfect matchings M of H. In particular, $|M_e \cap \delta_H(S)| = 1$ for all $e \in E(H)$ if and only if $w(\delta_H(S)) = |E(H)|$ holds. Thus, $S \subseteq V(H)$ defines a tight cut if and only if $r \nmid |S|$ and $w(\delta_H(S)) = |E(H)|$. By Corollary 10.4.7 in [Grötschel et al., 1988], we can solve $\min\{w(\delta_H(S)) : S \subseteq V(H), |S| \not\equiv 0 \bmod r\}$ in polynomial time. However, we want to find a non-trivial tight cut, thus we have to demand that $|S| \geq 2$ and $|S| \leq |V(H)| - 2$. We show that also the optimization problem

$$\min\{w(\delta_H(S)) : S \subseteq V(H), 2 \leq |S| \leq |V(H)| - 2, |S| \not\equiv 0 \bmod r\} \tag{4.7}$$

is polynomial-time solvable.

Therefore, let $A, B \subseteq V(H)$ be disjoint sets of vertices. The family $\mathcal{C}(A, B) := \{S \subseteq V(H) : A \subseteq S \subseteq V(H) \setminus B\}$ has the property that for every $S, T \in \mathcal{C}(A, B)$ also $S \cap T$ and $S \cup T$ lie in $\mathcal{C}(A, B)$. Such a family of sets is called a lattice family in [Grötschel et al., 1988]. Again, by Corollary 10.4.7 in [Grötschel et al., 1988] applied to $\mathcal{C}(A, B)$, we can solve $\min\{w(\delta_H(S)) : S \in \mathcal{C}(A, B), |S| \not\equiv 0 \bmod r\}$ in polynomial time for every pair of fixed sets $A, B \subseteq V(H)$.

Now, problem (4.7) can be solved by calculating for all disjoint subsets A, B of $V(H)$ with $|A| = |B| = 2$ an optimal solution $S_{A,B}$ to the optimization problem $\min\{w(\delta_H(S)) : S \in \mathcal{C}(A, B), |S| \not\equiv 0 \bmod r\}$, and choosing a set

$$S^* := \operatorname{argmin}\{w(\delta_H(S_{A,B})) : A, B \subseteq V(H), A \cap B = \emptyset, |A| = |B| = 2\}.$$

If $w(S^*) = |E(H)|$, then $\delta_H(S^*)$ is a non-trivial tight cut in H. Otherwise, $w(S^*) > |E(H)|$, and H contains only trivial tight cuts. As there are $\mathcal{O}(|V(H)|^4)$

subsets $A, B \subseteq V(H)$ with $|A| = |B| = 2$, this algorithm runs in polynomial time. In total, we get the following result.

Theorem 4.64. *Let H be an r-uniform, matching covered, balanced hypergraph. There exists a polynomial time algorithm that either outputs a non-trivial tight cut $\delta_H(S)$ or concludes that H has only trivial tight cuts.*

It seems like the above method for finding non-trivial tight cuts in balanced uniform hypergraphs might be generalized to the problem of finding non-trivial separating cuts in uniform hypergraphs. This is not the case. The first problem that arises is that it is $\mathcal{NP}$-hard to find for some fixed hyperedge a perfect matching containing it. Even if we assume that we have not only given a uniform hypergraph H as input but also a set of perfect matchings $\{M_e : e \in E(H)\}$ with $e \in M_e$ for all $e \in E(H)$, it is not clear how to find a non-trivial separating cut or non-trivial tight cut in H. Using the method described above we can find in polynomial time a non-trivial cut $\delta_H(S)$ with $|\delta_H(S) \cap M_e| = 1$ for all $e \in E(H)$ or decide that none exists. In the first case, we conclude that $\delta_H(S)$ is a separating cut. In the latter case it is still possible that H has a non-trivial separating cut but we have chosen the wrong perfect matchings. Indeed, it is not known if one can decide in polynomial time whether or not a brick, which is a non-bipartite graph without non-trivial tight cuts, has a separating cut.

Chapter 5

Flows in Directed Hypergraphs

Flow problems as the maximum s, t-flow or minimum cost flow problem in directed graphs are one of the best studied objects in combinatorial optimization. Though those types of problems can often be formulated as linear programs, and thus be solved by any linear programming solver, their combinatorial properties are not only of theoretical interest but also lead to various highly efficient algorithms. These combinatorial algorithms outperform general purpose linear programming solvers on large test instances, see [Kovács, 2015] for a current experimental study of various minimum cost flow algorithms.

Network flows are successfully used in many applications. However, sometimes it is more appropriate to use directed hypergraphs as a modeling tool. This is the case in vehicle rotation planning of Intercity-Express trains in Germany, see [Borndörfer et al., 2011] and [Borndörfer et al., 2012]. Building on the hypergraph model in this application we give a network simplex type algorithm for the minimum cost flow problem on directed hypergraphs in this chapter.

First, we give an overview about the literature on directed hypergraphs focusing on flow and path problems. The relation between those two types of problems is investigated in Section 5.2, where we look at two different integer programming formulations for the maximum s, t-flow problem. One formulation with a variable for every hyperarc and another one with a variable for every path. Both formulations are equivalent in the case of directed graphs. The path-based formulation was considered in a more general setting by Hoffman in [Hoffman, 1974], where he proves that it leads to an integral primal-dual pair of linear programs. In contrast to this result, the linear programming relaxation of the hyperarc-based integer program is not integral in general. We conclude Section 5.2 by giving optimality conditions for maximum s, t-flows and minimum cost flows in directed hypergraphs based on the idea of residual networks, which plays an important role in network flow problems on directed graphs.

In Section 5.3 we state the main result of this chapter, which is a combinatorial algorithm for the minimum cost flow problem on directed hypergraphs, where we use the hypergraph model of [Borndörfer et al., 2011]. This algorithm can be seen as a generalization of the network simplex algorithm. Using the directed hypergraph notion of [Borndörfer et al., 2011] has the advantage that it is possible to generalize a lot of the combinatorial steps in the network simplex algorithm to the hypergraphic setting. A short preliminary version of the results presented in Section 5.3 appeared in [Beckenbach, 2018].

5.1 Literature Overview

The literature on directed hypergraphs is diverse and there is no consistent definition of a directed hypergraph. A widely accepted one was given by Gallo, Pallottino, and Nguyen [Gallo et al., 1993].

Definition 5.1 (Directed hypergraph)**.** A *directed hypergraph H* is a pair consisting of a finite set of vertices $V(H)$ and a family of hyperarcs $E(H)$, where a *hyperarc* is an ordered pair $e = (t(e), h(e))$ of disjoint subsets of $V(H)$ such that not both subsets are empty. For a hyperarc e we denote by $t(e)$ the first element of e and call it the *tail* of e, and we denote by $h(e)$ the second element and call it the *head* of e.

If $|t(e)| \geq 2$ or $|h(e)| \geq 2$, we say that e is a *proper hyperarc*. If $|t(e)| = |h(e)| = 1$, then we call e an *arc* and write $e = (v, w)$ instead of $e = (\{v\}, \{w\})$. A hyperarc is called a *B-arc* if the size of its head is one and an *F-arc* if the size of its tail is one. A *B-graph* is a directed hypergraph with only B-arcs as hyperarcs. Similar, an *F-graph* is a directed hypergraph with only F-arcs.

A hyperarc is allowed to have an empty tail or an empty head, however, at least one of the two sets has to be non-empty. Furthermore, an arc is both a B-arc and an F-arc, and a directed graph is both a B-graph and an F-graph.

Figure 5.1 shows an example of a directed hypergraph on six vertices with four proper hyperarcs where two of them are F-arcs and one is a B-arc.

Directed hypergraphs are for example used in the analysis of metabolic networks. A biochemical reaction transforming substrates $A_1, \ldots, A_k$ into $B_1, \ldots, B_l$ can be modeled by an hyperarc whose tails are $A_1, \ldots, A_k$ and whose heads are $B_1, \ldots, B_l$, see [Klamt et al., 2009].

Another generalization of directed graphs to hypergraphs is motivated by an application to railway vehicle rotation planning, see [Borndörfer et al., 2011].

Definition 5.2 (Graph-based directed hypergraph)**.** Let D be a (standard) directed graph. A directed hypergraph H based on the directed graph D has the same set of vertices as D, and a set $E(H) \subseteq 2^{E(D)}$ of hyperarcs, where a hyperarc h consists of a set of vertex disjoint arcs of D. A directed hypergraph is called *graph-based* if it is based on some directed graph.

A graph-based hypergraph can be seen as a special kind of directed hypergraph in which each hyperarc has the same number of vertices in its tail as in its head. Namely, for a hyperarc e in a graph-based directed hypergraph we define its tail set

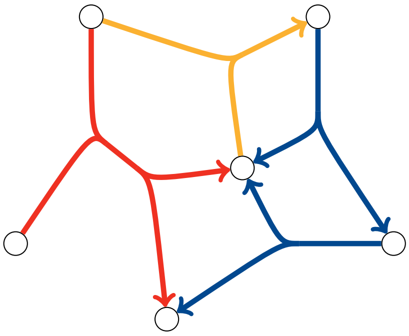


Figure 5.1: A directed hypergraph with one B-arc and two F-arcs.

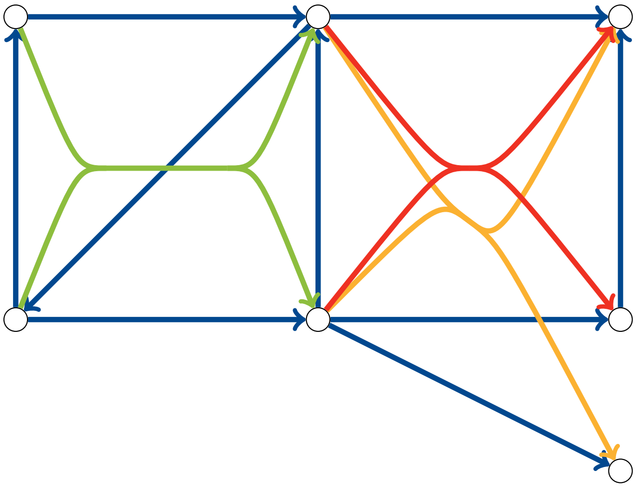


Figure 5.2: A graph-based directed hypergraph.

by $t(e) := \{v \in V(H) : \exists w \in V(H) \text{ with } (v,w) \in e\}$ and its set of head vertices by $h(e) := \{v \in V(H) : \exists w \in V(H) \text{ with } (w,v) \in e\}$. With this notation $(t(e), h(e))$ corresponds to a hyperarc according to Definition 5.1 with $|t(e)| = |h(e)|$. On the other hand, if H is a directed hypergraph such that $|t(e)| = |h(e)|$ holds for all hyperarcs e, then H can be seen as a directed hypergraph based on the complete directed graph on the vertex set $V(H)$. It turns out that it has advantages to keep the underlying digraph in mind when looking at flow problems.

Reuther shows in his dissertation [Reuther, 2017] the strength of graph-based hypergraphs in practice. A lot of requirements in railway vehicle rotation planning can be expressed by directed hyperarcs. Based on this hypergraph model Reuther developed a software tool that was successfully used at DB Fernverkehr AG to optimize Intercity-Express rotations for the German high-speed railway network.

There are different notions of paths and connectivity in directed hypergraphs. The simplest definition of a path is a straightforward generalization of the one for directed graphs.

Definition 5.3 (Path)**.** A *path* in a directed hypergraph H is an alternating sequence of vertices and hyperarcs $(v_1, e_1, v_2, e_2, \dots, e_k, v_{k+1})$ such that $v_1 \in t(e_1)$, $v_{k+1} \in h(e_k)$, and $v_i \in h(e_{i-1}) \cap t(e_i)$ for $i = 2, \dots, k$. A path is called *simple* if all vertices and hyperarcs are distinct. A vertex s is *connected* to a vertex t in H if there is a path starting at s and ending at t.

This kind of connectivity reduces to the digraph case by adding an auxiliary vertex v_e for every hyperarc e and arcs (v, v_e) and (v_e, w) for all $v \in t(e)$, $w \in h(e)$. In this way, a vertex s is connected to another vertex t in a directed hypergraph if and only if there is a directed path from s to t in the auxiliary digraph.

There are other definitions of connectivity in directed hypergraphs. For example, in B-graphs one usually demands that all tail vertices of a hyperarc have to be visited before one can use this hyperarc to visit its head vertex. Formally, we have the following definition, see for example [Thakur and Tripathi, 2009], where also other variants of connectivity are discussed.

Definition 5.4 (B-hyperpath, B-connectivity)**.** Given two vertices s and t in a B-graph H, we say that t is *B-connected* to s if $t = s$ or there exists a hyperarc e such that $h(e) = \{t\}$ and every $v \in t(e)$ is B-connected to s. A *B-hyperpath* from s to t is a minimal B-graph P with the properties that $V(P) \subseteq V(H)$, $E(P) \subseteq E(H)$, and t is B-connected to s.

Gallo, Longo, Pallottino, and Nguyen give a polynomial time algorithm that finds all vertices that are B-connected to some vertex s and show that it is possible to find a minimum weight B-hyperpath from s to t in polynomial time if a special recursively defined weight function on the set of B-hyperpaths is used, see [Gallo et al., 1993]. A formal definition of these kind of weight functions can be found in [Ausiello et al., 2001]. On the other hand, finding a B-hyperpath from s to t containing as few hyperarcs as possible is $\mathcal{NP}$-hard, see [Ausiello and Laura, 2017].

There are also different possibilities to define a cut in a directed hypergraph. We use the following one.

Definition 5.5 (Directed cut)**.** Let H be a directed hypergraph, and $S \subseteq V(H)$ be a non-empty set of vertices. The *out-cut* induced by S is defined by

$$\delta_H^{out}(S) := \{e \in E(H) : t(e) \cap S \neq \emptyset, h(e) \setminus S \neq \emptyset\},$$

and the *in-cut* is defined by

$$\delta_H^{in}(S) := \{e \in E(H) : h(e) \cap S \neq \emptyset, t(e) \setminus S \neq \emptyset\}.$$

For $S = \{v\}$ we just write $\delta_H^{out}(v)$ and $\delta_H^{in}(v)$. If no confusion might occur we drop H from the index.

Now, we turn to the minimum cost flow problem on directed hypergraphs.

Definition 5.6 (Minimum cost hyperflow)**.** Given a directed hypergraph H, costs $c : E(H) \to \mathbb{Q}$, hyperarc capacities $u : E(H) \to \mathbb{Q}_{\geq 0}$, and demands $d : V(H) \to \mathbb{Q}$ on the vertices, a *minimum cost hyperflow* is a solution to the following linear program:

$$\begin{aligned} & \min \sum_{e \in E(H)} c(e) z_e \\ \text{s.t.} \sum_{e \in \delta_H^{in}(v)} z_e - & \sum_{e \in \delta_H^{out}(v)} z_e = d(v) \ \forall v \in V(H) \\ 0 \leq & z_e \leq u(e) \ \forall e \in E(H). \end{aligned} \tag{5.1}$$

If $u \equiv \infty$ we call the minimum cost hyperflow problem *uncapacitated.*

There exists a polynomial time primal-dual algorithm for the uncapacitated minimum cost hyperflow problem on so-called gain-free B-graphs for non-negative demands $d \geq 0$, see [Jeroslow et al., 1992]. In this particular case the linear program (5.1) is totally dual integral. In general, finding an integral minimum cost hyperflow in a directed hypergraph is $\mathcal{NP}$-hard. For example, every bounded integer program can be transformed into a minimum cost hyperflow problem on an auxiliary B-graph with capacities on the hyperarcs, see [Cambini et al., 1992] (the preprint version of [Cambini et al., 1997]) for details.

Cambini, Gallo, and Scutellà give in [Cambini et al., 1997] a "network simplex" algorithm for (5.1) by interpreting all simplex operations combinatorially. They specify no simplex rule and there is none known requiring a polynomial number of pivots, thus their method is not polynomial. The algorithm developed in Section 5.3 is similar to this one. However, we heavily use that we work on graph-based directed hypergraphs, which simplifies a lot of the steps in the algorithm of [Cambini et al., 1997].

For more results on directed hypergraphs we refer to the recent survey by Ausiello and Laura [Ausiello and Laura, 2017] and to [Gallo et al., 1993], where the latter focuses on connectivity problems.

5.2 Paths and Flows in Directed Hypergraphs

In a directed graph an s,t-flow can either be seen as a flow along some arcs or along directed s,t-paths leading to two different linear programming formulations of the maximum s,t-flow problem. It turns out that these two approaches are equivalent and that the resulting linear programs are totally dual integral. In the hypergraph setting this is not the case as we show in this section.

First, we consider a linear programming formulation based on paths. Hoffman gives in [Hoffman, 1974] a general max-flow-min-cut theorem including the well known result of Ford and Fulkerson [Ford and Fulkerson, 1956] stating that the maximum value of an s,t-flow in a directed graph with arc capacities u is equal to the minimum capacity of an s,t-cut. Hoffman's theorem can be applied to the directed hypergraph setting by assigning a variable to each simple s,t-path. Therefore, we denote by $\mathcal{E}_{s,t}$ the set of all simple paths from s to t.

Definition 5.7 (s,t-path flow)**.** Given a directed graph H, two distinct vertices $s,t \in V(H)$, and capacities $u : E(H) \to \mathbb{Z}_{\geq 0}$ on the hyperarcs, an *s,t-path flow* is a function $y : \mathcal{E}_{s,t} \to \mathbb{Z}_{\geq 0}$ such that $\sum_{F \ni e} y_F \leq u(e)$ for all $e \in E(H)$. The *value* of an s,t-path flow y is $\sum_{F \in \mathcal{E}_{s,t}} y_F$.

An s,t-path flow can be seen as a multi-set of s,t-paths such that every hyperarc e is contained in at most $u(e)$ paths. For these kind of s,t-flows a generalization of the max-flow-min-cut theorem holds.

Theorem 5.8 ([Hoffman, 1974])**.** *Given a directed hypergraph H, two distinct vertices $s,t \in V(H)$, and capacities $u : E(H) \to \mathbb{Z}_{\geq 0}$, the following two linear programs have optimal solutions that are integral:*

$$(5.2) \qquad \begin{aligned} \max \sum_{F \in \mathcal{E}_{s,t}} & y_F \\ s.t. \sum_{F \ni e} & y_F \leq u(e) \ \forall \ e \in E(H) \\ & y_F \geq 0 \ \forall F \in \mathcal{E}_{s,t} \end{aligned} \qquad\qquad (5.3) \qquad \begin{aligned} \min \sum_{e \in E} & u(e) x_e \\ s.t \sum_{e \in F} & x_e \geq 1 \ \forall F \in \mathcal{E}_{s,t} \\ & x_e \geq 0 \ \forall e \in E(H) \end{aligned}$$

An optimal solution to the linear program on the left hand side is an s,t-path flow of maximum value. On the other hand, every integral optimal solution of the linear program on the right hand side has entries in $\{0,1\}$, and such a solution corresponds to a set of hyperarcs intersecting every s,t-path.

Definition 5.9. Given a directed hypergraph H, and two distinct vertices s,t, a set $C \subseteq E(H)$ of hyperarcs is an *s,t-path cut* if $C \cap F \neq \emptyset$ for all $F \in \mathcal{E}_{s,t}$.

With this notation we can restate Theorem 5.8:

> The maximum value of an s,t-path flow is equal to the minimum capacity of an s,t-path cut in a directed hypergraph.

Viewing s,t-flows as flows along hyperarcs leads to a different s,t-flow concept.

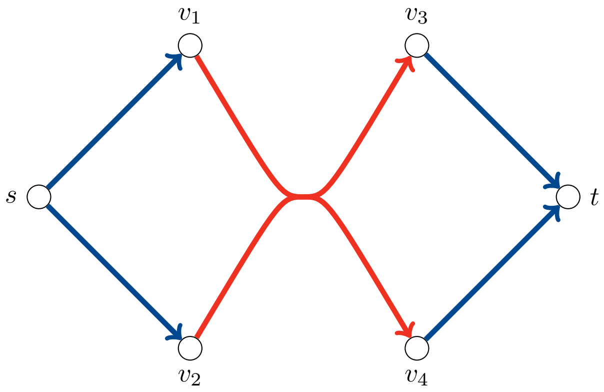


Figure 5.3: A directed hypergraph with a maximum s,t-arc flow of value two and a maximum s,t-path flow of value one.

Definition 5.10. Given a directed graph H, two distinct vertices $s,t \in V(H)$, and capacities $u : E(H) \to \mathbb{Q}_{\geq 0}$ on the hyperarcs, an *s,t-arc flow* is defined as a vector $z \in \mathbb{Q}^{E(H)}$ such that

$$\sum_{e \in \delta^{in}(v)} z_e - \sum_{e \in \delta^{out}(v)} z_e = 0 \; \forall v \in V(H) \setminus \{s,t\}, \tag{5.4}$$

$$0 \leq z_e \leq u(e). \tag{5.5}$$

The *value* of z is $\sum_{e \in \delta^{in}(t)} z_e - \sum_{e \in \delta^{out}(t)} z_e$.

In the s,t-arc flow case no similar result to Theorem 5.8 holds. In particular, it is not the case that there exists a maximum s,t-flow that is integral if the capacities are integral.

If y is feasible to (5.2), then we define a vector $x \in \mathbb{Q}^{E(H)}_{\geq 0}$ by $x_e := \sum_{F \ni e} y_F$. With this definition (5.5) is satisfied. In the graph case x satisfies also (5.4) and the flow into t is equal to $\sum_{F \in \mathcal{E}_{s,t}} y_F$. However, in directed hypergraphs the vector x defined as above does not satisfy the flow conservation constraints (5.4).

On the other hand, every feasible solution to (5.4)-(5.5) in the digraph case can be written as the sum of incidence vectors of simple paths and directed cycles. As directed cycles do not contribute to the flow into the sink t we can cancel them and obtain a feasible solution to (5.2). Thus, (5.2) and (5.4)-(5.5) are equivalent on directed graphs.

The following example shows that on general directed hypergraphs s,t-path flows and s,t-arc flows are not equivalent. Furthermore, neither is dominated by the other as it is possible that the maximum value of an s,t-path flow is larger than the maximum value of an s,t-arc flow, or the other way around.

Example 5.11. Let H be a directed hypergraph with vertex set $\{s, t, v_1, v_2, v_3, v_4\}$ and hyperarcs (s, v_1), (s, v_2), $(\{v_1, v_2\}, \{v_3, v_4\})$, (v_3, t), (v_4, t), compare Figure 5.3. If all hyperarcs have capacity 1, then the optimal solution to (5.2) is one as every s, t-path contains the hyperarc $(\{v_1, v_2\}, \{v_3, v_4\})$, while an optimal arc flow of value two exists (one unit of flow on each hyperarc). Adjusting this example by adding a new vertex s' and an arc (s', s) with capacity one shows the arc-based LP can have a fractional optimal solution (in this case a flow of value 0.5 on all hyperarcs except of (s', s), which carries a flow of one unit).

On the other hand the hypergraph on $\{s, v_1, v_2, v_3, t\}$ with arcs (s, v_1), (v_1, v_2), (v_1, v_3), and hyperarc $(\{v_2, v_3\}, \{t\})$, all of capacity one, has an s, t-path flow of value one whereas the maximum value of an s, t-arc flow is 0.5.

When considering arc flows in directed hypergraphs a result similar to the max-flow-min-cut theorem does not hold. However, for graph-based directed hypergraphs a minimum cut gives at least an upper bound on the maximum flow value.

Theorem 5.12. *Let H be a graph-based directed hypergraph, $s, t \in V(H)$ be two distinct vertices, $u : E(H) \to \mathbb{Q}_{\geq 0}$ be capacities on the hyperarcs, and $S \subseteq V(H)$ be a set of vertices with $s \in S, t \notin S$. The maximum value of an s, t-arc flow is at most $\sum_{e \in \delta^{out}(S)} |t(e) \cap S| u(e)$.*

Proof. Let z be an s, t-arc flow, and $S \subseteq V(H)$ be a set of vertices with $s \in S$, $t \notin S$. We have to show that $z(\delta^{in}(t)) - z(\delta^{out}(t)) \leq \sum_{e \in \delta^{out}(S)} |t(e) \cap S| u(e)$ holds. As H is graph-based we have $|t(e)| = |h(e)|$ for all hyperarcs $e \in E(H)$. This implies that

$$\sum_{v \in V(H)} \Big(z(\delta^{in}(v)) - z(\delta^{out}(v)) \Big) = \sum_{e \in E(H)} (|h(e)| - |t(e)|)\, z_e = 0,$$

which shows that the flow going into the sink t is equal to the flow going out of the source s, i.e., the value of z is also equal to $z(\delta^{out}(s)) - z(\delta^{in}(s))$. Using this fact and the flow conservation (5.4) for all $v \in S \setminus \{s\}$ gives

$$\begin{aligned} z(\delta^{out}(s)) - z(\delta^{in}(s)) &= \sum_{v \in S} \Big(z(\delta^{out}(v)) - z(\delta^{in}(v)) \Big) \\ &= \sum_{e \in E(H)} (|t(e) \cap S| - |h(e) \cap S|)\, z_e \\ &\leq \sum_{e : |t(e) \cap S| > |h(e) \cap S|} (|t(e) \cap S| - |h(e) \cap S|)\, z_e \\ &\leq \sum_{e : |t(e) \cap S| > |h(e) \cap S|} |t(e) \cap S| z_e \leq \sum_{e \in \delta^{out}(S)} |t(e) \cap S| u(e). \end{aligned}$$

For the last inequality we use $z \leq u$ and the fact that $|t(e) \cap S| > |h(e) \cap S|$ implies that $t(e) \cap S \neq \emptyset$ and $h(e) \setminus S \neq \emptyset$, i.e., $e \in \delta^{out}(S)$. □

There are examples such that strict inequality holds in the previous theorem. If the hyperarc (s, v_1) in Figure 5.3 has capacity zero and all other (hyper)arcs capacity one, then the maximum value of an s, t-arc flow is zero whereas the minimum of $\sum_{e\in\delta^{out}(S)} |t(e) \cap S|u(e)$ is one.

A key observation in the directed graph case is that an s, t-flow in a digraph is optimal if and only if there exists no s, t-path in its residual graph. It is possible to define a similar condition for directed hypergraphs. Given a feasible arc flow we define a residual hypergraph in the same way as in the graph case.

Definition 5.13 (Residual hypergraph)**.** Let H be a directed hypergraph with capacities $u : E(H) \to \mathbb{Q}_{\geq 0}$ on the hyperarcs, and $z \in \mathbb{Q}^{E(H)}$ be a vector with $0 \leq z_e \leq u(e)$ for all $e \in E(H)$. For every hyperarc $e \in E(H)$ we define the *reverse hyperarc* $\overleftarrow{e}$ of e by $\overleftarrow{e} := (h(e), t(e))$. The *residual capacity* of a hyperarc $e \in E(H)$ with respect to z is $u_z(e) := u(e) - z(e)$ and that of its reverse hyperarc is $u_z(\overleftarrow{e}) := z(e)$. The *residual hypergraph* H_z is the directed hypergraph with vertex set $V(H)$ and hyperarcs $e \in E(H)$ with $u_z(e) > 0$ as well as $\overleftarrow{e}$ with $u_z(\overleftarrow{e}) > 0$ for $e \in E(H)$.

We characterize the optimality of an s, t-arc flow in terms of its residual hypergraph. Namely, an s, t-arc flow z is optimal if and only if it is not possible to ship a positive amount of flow from s to t in the residual hypergraph H_z such that flow conservation holds at all other vertices. This result is not very deep and can be proven similarly as in the graph case. We add it because we could not find any reference for it, and it seems that it has not been published before.

Theorem 5.14. *Let H be a directed hypergraph, s, t be two distinct vertices of H, $u : E(H) \to \mathbb{Q}_{\geq 0}$ be capacities on the hyperarcs. A vector $z \in \mathbb{Q}^{E(H)}$ is an optimal s, t-arc flow if and only if*

a) $0 \leq z_e \leq u(e)$ for all $e \in E(H)$, and

b) there does not exists a vector $f \in \mathbb{Z}_{\geq 0}^{E(H_z)}$ with $f(\delta^{in}(v)) = f(\delta^{out}(v))$ for all $v \in V(H_z) \setminus \{s, t\}$ and $f(\delta^{in}(t)) - f(\delta^{out}(t)) > 0$.

Proof. First, let $z \in \mathbb{Q}^{E(H)}$ be a vector with $0 \leq z \leq u$. We assume that a vector f as above exists and show that z is not optimal in this case.

We set $\alpha := \min\{\frac{u_z(e)}{f(e)} : f(e) > 0\}$, and define a new vector $x \in \mathbb{Q}^{E(H)}$ by $x(e) := z(e) + \alpha \cdot (f(e) - f(\overleftarrow{e}))$. By the choice of α we have

$$\begin{aligned} x(e) &\leq z(e) + \alpha f(e) \leq z(e) + u_z(e) = u(e), \text{ and} \\ x(e) &\geq z(e) - \alpha f(\overleftarrow{e}) \geq z(e) - u_z(\overleftarrow{e}) = 0. \end{aligned}$$

For every vertex $v \in V(H) \setminus \{s,t\}$ we get

$$\begin{aligned} x(\delta^{in}_H(v)) - x(\delta^{out}_H(v)) &= z(\delta^{in}_H(v)) - z(\delta^{out}_H(v)) + \sum_{e \in E(H), v \in h(e)} \alpha\,(f(e) - f(\overleftarrow{e})) \\ &\quad - \sum_{e \in E(H), v \in t(e)} \alpha\,(f(e) - f(\overleftarrow{e})) \\ &= \alpha\left(f(\delta^{in}_{H_z}(v)) - f(\delta^{out}_{H_z}(v))\right) = 0 \end{aligned}$$

Similar, for $v = t$ we get

$$\begin{aligned} x(\delta^{in}_H(t)) - x(\delta^{out}_H(t)) &= z(\delta^{in}_H(t)) - z(\delta^{out}_H(t)) + \alpha\left(f(\delta^{in}_{H_z}(t)) - f(\delta^{out}_{H_z}(t))\right) \\ &> z(\delta^{in}_H(t)) - z(\delta^{out}_H(t)). \end{aligned}$$

Thus, x is an s,t-arc flow of larger value than z.

On the other hand, suppose z with $0 \le z \le u$ is not optimal and x is an s,t-arc flow of larger value. We define a vector $f \in \mathbb{Q}^{E(H_z)}_{\ge 0}$ by $f(e) := x(e) - z(e)$ if $x(e) > z(e)$, $f(\overleftarrow{e}) := z(e) - x(e)$ if $x(e) < z(e)$, and otherwise f takes value zero. Observe that f is well defined as $x(e) > z(e)$ implies that $u_z(e) > 0$ and $x(e) < z(e)$ implies $u_z(\overleftarrow{e}) > 0$.

For every $v \in V(H)$ we have

$$\begin{aligned} f(\delta^{in}_{H_z}(v)) - f(\delta^{out}_{H_z}(v)) &= \sum_{\substack{e: v \in h(e),\\ x(e) > z(e)}} (x(e) - z(e)) + \sum_{\substack{e: v \in t(e),\\ x(e) < z(e)}} (z(e) - x(e)) \\ &\quad - \sum_{\substack{e: v \in t(e),\\ x(e) > z(e)}} (x(e) - z(e)) - \sum_{\substack{e: v \in h(e),\\ x(e) < z(e)}} (z(e) - x(e)) \\ &= \sum_{e: v \in h(e)} (x(e) - z(e)) - \sum_{e: v \in t(e)} (x(e) - z(e)) \\ &= x(\delta^{in}_H(v)) - x(\delta^{out}_H(v)) - z(\delta^{in}_H(v)) + z(\delta^{out}_H(v)). \end{aligned}$$

Thus, $f(\delta^{in}_{H_z}(v)) - f(\delta^{out}_{H_z}(v)) = 0$ for $v \ne s,t$, and $f(\delta^{in}_{H_z}(t)) - f(\delta^{out}_{h_z}(t)) > 0$. Multiplying f by some scaler such that every entry becomes an integer gives a vector as desired. □

A similar condition can be given for the optimality of a minimum cost flow in a directed hypergraph. Therefore, we have to define a cost function on the residual hypergraph. Given a vector $z \in \mathbb{Q}^{E(H)}$ with $0 \le z \le u$, the *residual cost* of $e \in E(H)$ is $c_z(e) := c(e)$ and that of $\overleftarrow{e}$ is $c_z(\overleftarrow{e}) = -c(e)$.

Theorem 5.15. *Let H be a directed hypergraph, $d : V(H) \to \mathbb{Q}$ be demands on the vertices, $u : E(H) \to \mathbb{Q}_{\ge 0}$ be capacities on the hyperarcs, and $c : E(H) \to \mathbb{Q}$*

be costs on the hyperarcs. A vector $x \in \mathbb{Q}^{E(H)}$ with $0 \le x \le u$ is a minimum cost hyperflow if and only if

a) $x(\delta^{in}(v)) - x(\delta^{out}(v)) = d(v)$ for all $v \in V(H)$, and

b) there does not exists a non-zero vector $f \in \mathbb{Z}_{\ge 0}^{E(H_z)}$ with $f(\delta^{in}(v)) = f(\delta^{out}(v))$ for all $v \in V(H_z)$ and $\sum_{e \in E(H_z)} c_z(e) f(e) < 0$.

Proof. If there exists a non-zero vector $f \in \mathbb{Z}_{\ge 0}^{E(H_z)}$ with $f(\delta^{in}(v)) = f(\delta^{out}(v))$ for all $v \in V(H_z)$ we can show as in the previous proof that $x \in \mathbb{Q}^{E(H)}$ with $x(e) := z(e) + \alpha \cdot (f(e) - f(\overleftarrow{e}))$, where $\alpha := \min\{\frac{u_z(e)}{f(e)} : f(e) > 0\}$, is a feasible flow with respect to d and u. The cost of this flow is equal to

$$\begin{aligned} &\sum_{e \in E(H)} c(e)z(e) + \alpha \sum_{e \in E(H)} c(e)\,(f(e) - f(\overleftarrow{e})) \\ = &\sum_{e \in E(H)} c(e)z(e) + \alpha \sum_{e \in E(H_z)} c_z(e)f(e) \\ < &\sum_{e \in E(H)} c(e)z(e) \end{aligned}$$

On the other hand, if x is a feasible flow of smaller cost than z, then we can define f as in the second part of the previous proof. □

In the digraph case, the vectors f in Theorem 5.14 and Theorem 5.15 can be restricted to incidence vectors of s, t-paths in the first case and cycles in the latter. In general directed hypergraphs it is not easy to characterize the "minimal" elements that we have to consider in both theorems. However, on graph-based hypergraphs we can exploit the underlying digraph structure.

Definition 5.16. Let H be a graph-based directed hypergraph. A *hypercircuit* in H is a function $x : E(H) \to \mathbb{Z}_{\ge 0}$ such that $x(\delta_H^{in}(v)) = x(\delta_H^{out}(v))$ for all $v \in V(H)$ and $x(e) > 0$ for at least one $e \in E(H)$. A hypercircuit is *elementary* if there does not exist a hypercircuit x' with $x'(e) \le x(e)$ for all $e \in E(H)$ and $x'(e) < x(e)$ for at least one $e \in E(H)$.

Given two vertices $s, t \in V(H)$, we call a function $x : E(H) \to \mathbb{Z}_{\ge 0}$ with $x(\delta_H^{in}(v)) = x(\delta_H^{out}(v))$ for all $v \in V(H) \setminus \{s, t\}$ and $x(\delta_H^{in}(t)) - x(\delta_H^{out}(t)) > 0$ an *s, t-hyperpath*. An s, t-hyperpath x is *elementary* if there do not exist an s, t-hyperpath x' with $x'(e) \le x(e)$ for all $e \in E(H)$ and $x'(e) < x(e)$ for at least one $e \in E(H)$.

An elementary hypercircuit in a directed graph is just a simple directed cycle and an elementary s, t-hyperpath is a simple directed s, t-path. In the remainder of this section, if we speak of a cycle in a directed graph, we always mean a simple directed cycle, and an s,t-path in a directed graph is a simple directed s,t-path. Furthermore, we slightly abuse notation and write $a \in E(H)$ if $\{a\} \in E(H)$ for an arc a of the underlying digraph of H.

Using the structure of the underlying digraph of a graph based directed hypergraph, we show that hypercircuits consist of a union of cycles "linked" by hyperarcs.

Theorem 5.17. *Let H be a directed hypergraph based on a directed graph D. A vector $x \in \mathbb{Z}_{\geq 0}^{E(H)}$ is a hypercircuit if and only if*

$$x = \chi_{C_1} + \ldots + \chi_{C_l} + z_1 \cdot (\chi_{h_1} - \sum_{a \in h_1} \chi_a) + \ldots + z_k \cdot (\chi_{h_k} - \sum_{a \in h_k} \chi_a),$$

where $l \geq 1$, $k \geq 0$ are integers, $C_1, \ldots, C_l$ are cycles in D, $h_1, \ldots, h_k \in E(H)$ are proper hyperarcs, and $z_1, \ldots, z_k \in \mathbb{N}$ are natural numbers such that each $a \in E(D)$ is covered by at least $\sum_{i \in [k]: a \in h_i} z_i$ cycles of $C_1, \ldots, C_l$.

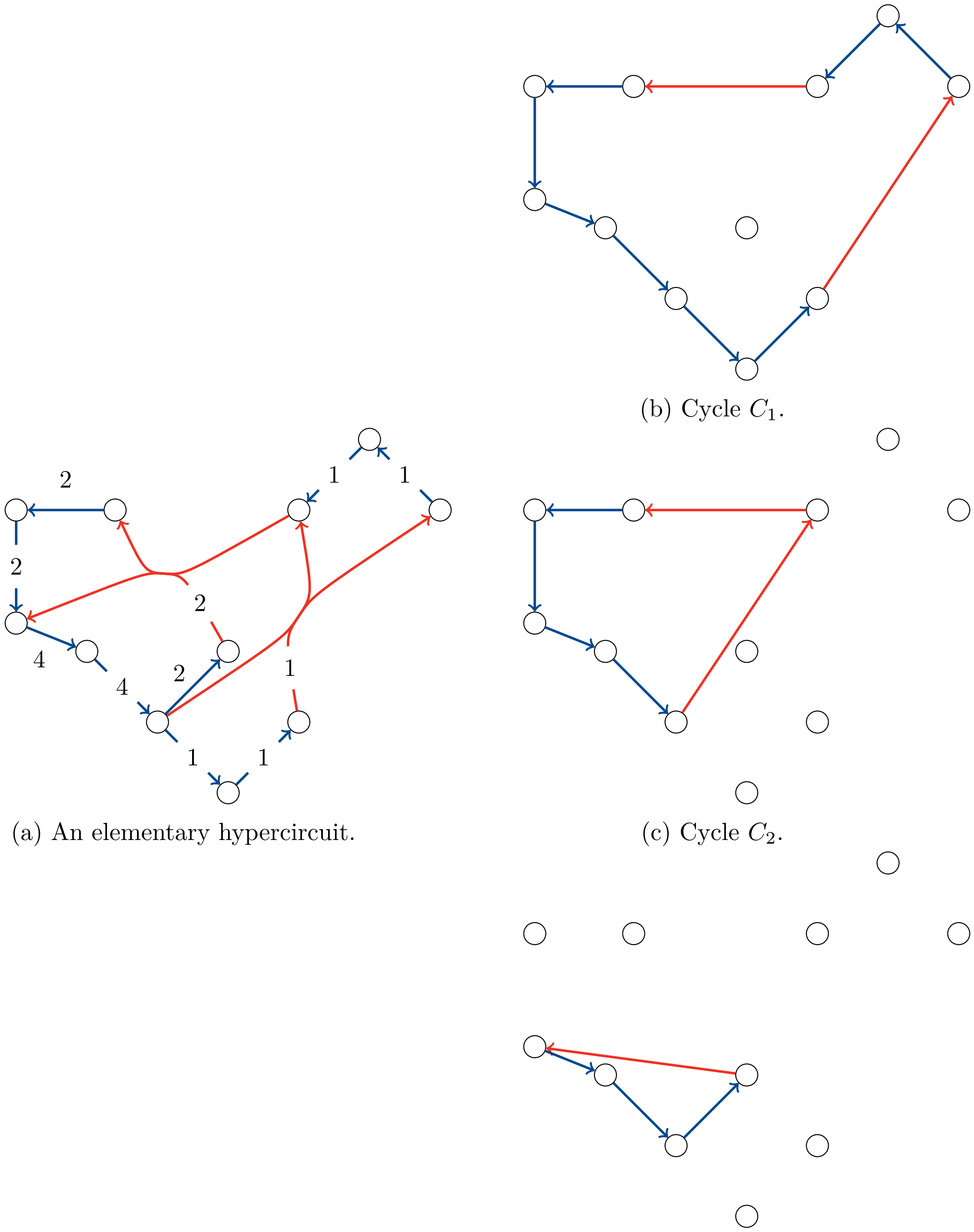
Proof. Let x be a hypercircuit in H and $\{h_1, \ldots, h_k\}$ be the set of proper hyperarcs with positive x-value. For every $a \in E(D)$ we set $x(a) := x(\{a\})$ if $\{a\} \in E(H)$ and $x(a) := 0$ otherwise. We define a vector $y \in \mathbb{Z}_{\geq 0}^{E(D)}$ by $y(a) := x(a) + \sum\limits_{i \in [k]: a \in h_i} x(h_i)$ for all $a \in E(D)$. The vector y satisfies $y(\delta_D^{in}(v)) = y(\delta_D^{out}(v))$ for all $v \in V(D)$, thus it is a circulation of D, and there exist cycles $C_1, \ldots, C_l$ such that $y = \sum_{i=1}^{l} \chi_{C_i}$. Setting $z_i := x(h_i)$ we can write x as

$$x = \chi_{C_1} + \ldots + \chi_{C_l} + z_1 \cdot (\chi_{h_1} - \sum_{a \in h_1} \chi_a) + \ldots + z_k \cdot (\chi_{h_k} - \sum_{a \in h_k} \chi_a).$$

Using $y(a) \geq \sum\limits_{i \in [k]: a \in h_i} x(h_i)$ we get that each $a \in E(D)$ is covered by at least $\sum\limits_{i \in [k]: a \in h_i} z_i$ cycles of $C_1, \ldots, C_l$.

For the other direction, let $C_1, \ldots, C_l$ be cycles, $h_1, \ldots, h_k \in E(H)$ be proper hyperarcs, and $z_1, \ldots, z_k \in \mathbb{N}$ be natural numbers with the property stated in the theorem. For every $v \in V(H)$ we have to show that $x(\delta^{in}(v)) = x(\delta^{out}(v))$. This follows from the fact that $\chi_{C_i}(\delta^{in}(v)) = \chi_{C_i}(\delta^{out}(v))$ for $i \in [l]$ and that each h_j is the disjoint union of the arcs $a \in h_j$ for $j \in [k]$. It remains to show that x is non-negative. By construction, $x(h_j) = z_j \geq 1$ for $j \in [k]$. For every $a \in E(D)$ we get that $x(a) = m(a) - \sum_{i \in [k]: a \in h_i} z_i$, where $m(a)$ is the number of times a is covered by $C_1, \ldots, C_l$. By assumption, $m(a) \geq \sum_{i \in [k]: a \in h_i} z_i$, and thus $x(a) \geq 0$. □

In a digraph the elementary hypercycles only take the values 0 and 1, thus they are sets of arcs. If we have proper hyperarcs of size at least two, then there might exist elementary hypercircuits that do not only take the values 0 and 1. An example of



(a) An elementary hypercircuit.

(b) Cycle C_1.

(c) Cycle C_2.

(d) Cycle C_3.

Figure 5.4: A decomposition of a hypercircuit into cycles.

such a hypercircuit is depicted in Figure 5.4a. Figure 5.4 shows the cycles occurring in a decomposition as in Theorem 5.17 where cycle C_3 occurs two times.

Similar as in the case of hypercircuits, we show that s,t-hyperpaths correspond to a set of s,t-paths and cycles linked by hyperarcs.

Theorem 5.18. *Let H be a directed hypergraph based on a directed graph D, and s, t be two distinct vertices of H. A vector $x \in \mathbb{Z}_{\geq 0}^{E(H)}$ is an s,t-hyperpath if and only if*

$$x = \chi_{P_1} + \ldots + \chi_{P_l} + \chi_{C_1} + \ldots + \chi_{C_m} + z_1 \cdot (\chi_{h_1} - \sum_{a \in h_1} \chi_a) + \ldots + z_k \cdot (\chi_{h_k} - \sum_{a \in h_k} \chi_a)$$

where $l \geq 1$, $m, l \geq 0$ are integers, $P_1, \ldots, P_l$ are s,t-paths in D, $C_1, \ldots, C_m$ are cycles in D, $h_1, \ldots, h_k \in E(H)$ are proper hyperarcs, and $z_1, \ldots, z_k \in \mathbb{N}$ are natural numbers such that each $a \in E(D)$ is covered by at least $\sum_{i \in [k]: a \in h_i} z_i$ paths and cycles of $P_1, \ldots, P_l, C_1, \ldots, C_m$.

Proof. Let x be an s,t-hyperpath in H and $\{h_1, \ldots, h_k\}$ be the set of proper hyperarcs with positive x-value. For every arc $a \in E(D)$ we set $x(a) := x(\{a\})$ if $\{a\} \in E(H)$ and $x(a) := 0$ if $\{a\} \notin E(H)$. Then, we define a vector $y \in \mathbb{Z}^{E(D)}$ by $y(a) := x(a) + \sum_{i \in [k]: a \in h_i} x(h_i)$ for all $a \in E(D)$. The vector y satisfies $y(\delta_D^{in}(v)) = y(\delta_D^{out}(v))$ for all $v \in V(D) \setminus \{s, t\}$ and $y(\delta_D^{in}(t)) - y(\delta_D^{out}(t)) > 0$. This implies that there exist s,t-paths $P_1, \ldots, P_l$ and cycles $C_1, \ldots, C_m$ such that $y = \sum_{i=1}^{l} \chi_{P_i} + \sum_{i=1}^{m} \chi_{C_i}$. Setting $z_i := x(h_i)$ it follows that

$$x = \chi_{P_1} + \ldots + \chi_{P_l} + \chi_{C_1} + \ldots + \chi_{C_m} + z_1 \cdot (\chi_{h_1} - \sum_{a \in h_1} \chi_a) + \ldots + z_k \cdot (\chi_{h_k} - \sum_{a \in h_k} \chi_a).$$

As $y(a) \geq \sum_{i \in [k]: a \in h_i} x(h_i)$, each $a \in E(D)$ is covered by at least $\sum_{i \in [k]: a \in h_i} z_i$ paths and cycles of $P_1, \ldots, P_l, C_1, \ldots, C_m$.

On the other hand, let x be of the form stated in the theorem. By similar arguments as in the proof of Theorem 5.17, the vector x is non-negative. It remains to show that flow conservation holds at all vertices distinct from s and t, and that there is a positive amount of flow going into t. For every vertex $v \in V(H)$ we have $\chi_{C_i}(\delta^{in}(v)) = \chi_{C_i}(\delta^{out}(v))$ for $i \in [m]$, $\chi_{h_j}(\delta^{in}(v)) - \sum_{a \in h_j} \chi_a(\delta^{in}(v)) = 0$, and $\chi_{h_j}(\delta^{out}(v)) - \sum_{a \in h_j} \chi_a(\delta^{out}(v)) = 0$ for $j \in [k]$ as each proper hyperarc h_j is the disjoint union of its underlying arcs $a \in h_j$. This implies that

$$x(\delta_H^{in}(v)) - x(\delta_H^{out}(v)) = \sum_{i=1}^{l} \left(\chi_{P_i} \left(\delta^{in}(v) \right) - \chi_{P_i} \left(\delta^{out}(v) \right) \right),$$

which is zero for all $v \in V(H) \setminus \{s, t\}$ and positive for $v = t$. Thus, x is an s,t-hyperpath. □

5.3 The Min Cost Hyperflow Problem on Graph-Based Directed Hypergraphs

In this section we describe a combinatorial algorithm for the minimum cost flow problem on graph-based directed hypergraphs that is based on the network simplex algorithm. In particular, it can be seen as a combinatorial variant of the primal simplex algorithm. Informally speaking, the basic idea of the primal simplex method for solving a linear program is to start with a feasible basic solution, which corresponds to a vertex of the associated polyhedron, and then find an improving direction to another basic solution, which is an adjacent vertex. In this way one goes from one vertex to another until one finds an optimal solution.

The minimum cost flow problem on a directed graph can be formulated as a linear program and thus be solved using the simplex algorithm. However, in this special setting one can exploit the underlying network structure leading to the so-called network simplex algorithm. It builds upon the fact that the basic solutions of the minimum cost flow problem when formulated as a linear program correspond to spanning trees. Starting with any spanning tree solution one can reach an adjacent basic solution by augmenting flow along cycles. During the network simplex algorithm one always maintains a spanning tree corresponding to a basic solution. Using special kind of spanning trees ensures that the algorithm always terminates in finite time with an optimal solution. For details we refer to Chapter 11 of [Ahuja et al., 1993]. There also exists a polynomial time variant of the network simplex algorithm, see [Orlin, 1997], in contrast to the simplex algorithm, where it is not known whether a simplex rule exists that leads to a polynomial running time.

In this section we develop a network simplex-type algorithm for the minimum cost hyperflow problem on graph-based directed hypergraphs. First, we characterize the basic solutions of the minimum cost hyperflow problem by showing that every such solution corresponds to a spanning forest and a small set of proper hyperarcs together with a special matrix describing the interaction between the trees of the forest and the proper hyperarcs in the basis. Building upon this characterization we develop a primal network simplex algorithm in the second subsection. We focus on the uncapacitated case to simplify the presentation and keep the focus on the main ideas. However, our algorithm can also be adapted to the capacitated case.

5.3.1 Basis Matrices of the Minimum Cost Hyperflow Problem

In this subsection we look at the structure of so-called basic solutions to the minimum cost hyperflow problem (5.1). First, we review what a basic solution in a general linear program is and recall that the basic solutions to the minimum cost flow problem correspond to spanning trees. Afterwards, we show that a basic solution to the minimum hyperflow problem correspond to a spanning forests together with a special matrix.

The main observation that leads to the development of the simplex algorithm is that if a linear program of the form

$$\begin{aligned} \max\, c^t x \\ Ax = d \\ x \geq 0 \end{aligned}$$

is feasible and bounded, then it has an optimal solution that is a vertex of the polyhedron defined by $Ax = d, x \geq 0$. For a matrix $A \in \mathbb{Q}^{m \times n}$ and a set $B \subseteq [n]$ we denote by A_B the $(m \times |B|)$-submatrix of A restricted to the columns with indices in B. By slight abuse of notation, we denote for a column vector $x \in \mathbb{Q}^n$ and a set $B \subseteq [n]$ by x_B the vector in $\mathbb{Q}^B$ with $(x_B)_i = x_i$ for all $i \in B$, i.e., we restrict x to the rows with indices in B.

If A has rank r, $B \subseteq [n]$ has size r, and $A_B z = d$ has a unique solution, then $x^* \in \mathbb{Q}^n$ defined by $A_B x^*_B = d$ and $x^*_j = 0$ for all $j \in [n] \setminus B$ is called a *basic solution*, and A_B a *basis* of the system $Ax = d, x \geq 0$. Sometimes we also call B a basis if the context is clear. Observe that $B \subseteq [n]$ is a basis of $Ax = d, x \geq 0$ if $A_B x = d$ has a solution and the rank of A_B is equal to r.

Every vertex of the polyhedron $\{x \in \mathbb{Q}^n : Ax = d, x \geq 0\}$ is a basic solution but not the other way around because it is possible that a basic solution x^* has negative entries. A basic solution with only non-negative entries is called a *feasible basic solution*. Thus, the vertices of the polyhedron defined by $Ax = d, x \geq 0$ correspond exactly to the feasible basic solutions of $Ax = d, x \geq 0$.

We turn to the minimum cost flow problem and consider the basic solutions there. The minimum cost flow problem on a digraph D with costs $c \in \mathbb{Q}^{E(D)}$ and demands $d \in \mathbb{Q}^{V(D)}$ can be written as

$$\begin{aligned} \min\, c^t x \\ Ax = d \\ x \geq 0, \end{aligned}$$

where $A \in \{0, -1, 1\}^{V(D) \times E(D)}$ is the vertex-arc incidence matrix of D, i.e.,

$$A_{v,e} = \begin{cases} 1, & v = h(e) \\ -1, & v = t(e) \\ 0, & v \notin h(e) \cup t(e). \end{cases}$$

We assume that D is weakly connected as we can solve the minimum cost flow problem on each weakly connected component separately. Therefore, D has at least $|V(D)| - 1$ arcs and one can show that A has rank $|V(D)| - 1$. Furthermore, we assume that all the demands sum up to zero as otherwise there exists no solution to $Ax = d$. Now, the basic solutions to $Ax = d$ have the following special form, see for example Section 11.11 in [Ahuja et al., 1993].

Lemma 5.19. *Given a directed graph D and $d : V(D) \to \mathbb{Q}$ with $\sum_{v \in V(D)} d(v) = 0$, a set $B \subseteq E(D)$ is a basis of $Ax = d$ if and only if $D[B]$ is a spanning tree of D when seen as an undirected graph.*

This lemma is the starting point of the network simplex algorithm as it gives a combinatorial interpretation of the bases of the minimum cost flow problem. In particular, we can decide whether a set B of arcs is a basis by just looking at the subgraph induced by B and not considering any matrices.

In the remainder of this subsection we give a characterization of the bases of the minimum cost hyperflow problem. Let H be a hypergraph based on a directed graph D and let $A \in \{0, 1, -1\}^{V(H) \times E(H)}$ be its vertex-hyperarc incidence matrix, i.e.,

$$A_{v,e} = \begin{cases} 1, & v \in h(e) \\ -1, & v \in t(e) \\ 0, & v \notin h(e) \cup t(e). \end{cases}$$

With this definition, the inequalities of the form $x(\delta^{in}(v)) - x(\delta^{out}(v)) = d(v)$ for $v \in V(H)$ can be written as $Ax = d$, where $d : V(H) \to \mathbb{Q}$ are given demands with $\sum_{v \in V(H)} d(v) = 0$.

We assume without loss of generality that D is weakly connected and $\{e\} \in E(H)$ for all $e \in E(D)$. This implies that the rank of A is the same as the rank of the vertex-arc incidence matrix of D, which is $|V(D)| - 1$. Furthermore, there exists a solution to $Ax = d$ if and only if $A(D)x = d$ has a solution, where $A(D)$ denotes the vertex-arc incidence matrix of D. It is well known that for a weakly connected digraph D the system $A(D)x = d$ is solvable if and only if the entries of d sum up to zero. Therefore, $Ax = d$ has a feasible solution if and only if $\sum_{v \in V(H)} d(v) = 0$. In this case, $B \subseteq E(H)$ is a basis of $Ax = d$ if and only if $|B| = |V(D)| - 1$ and A_B has rank $|V(H)| - 1$.

For a set $B \subseteq E(H)$ of size $|V(H)| - 1$ we denote by $B_1 = \{e \in B : |e| = 1\}$ the set of all arcs and by $B_2 := B \setminus B_1$ the set of all proper hyperarcs of B. If B is a basis, then $D[\{e \in E(D) : \{e\} \in B_1\}]$ does not contain any cycles, and is therefore a forest having $|B_2| + 1$ connected components. If $B_2 \neq \emptyset$, this condition is not sufficient for B to be a basis.

Example 5.20. Figure 5.5 shows a subgraph of a directed hypergraph H restricted to a set of hyperarcs B. This set consists of two proper hyperarcs and 13 arcs. The arcs induce a forest with three connected components. However, B is not a basis

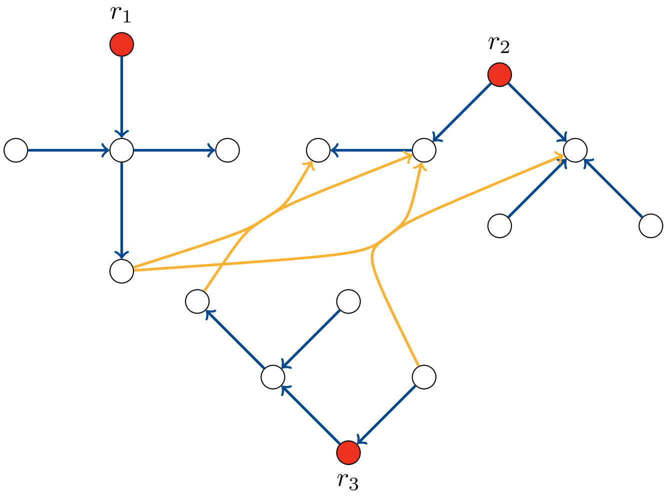


Figure 5.5: Set of hyperarcs B not forming a basis.

because the system $A_B x = d$ for

$$d(v) := \begin{cases} -1, & v = r_1 \\ 1, & v = r_2 \\ 0, & \text{otherwise} \end{cases}$$

has no solution.

If B is a set of hyperarcs of size $|V(H)| - 1$ such that $D[B_1]$ is a forest, then we choose for each of the $|B_2| + 1$ trees in $D[B_1]$ a root r, denote by R the set of those roots, and by $\{T_r\}_{r \in R}$ the set of the trees in $D[B_1]$. If B forms a basis, then we can send one unit of flow from each root r_1 to any other root $r_2 \in R \setminus \{r_1\}$, formally, the system $A_B x = d$ for d given by $d(r_1) = -1$, $d(r_2) = 1$ and $d(v) = 0$ for all other vertices has a solution $x \in \mathbb{Q}^{V(H)}$. We show that this condition together with the requirement that $D[B_1]$ is a forest consisting of $|B_2| + 1$ components gives a characterization of the basis matrices in the minimum cost hyperflow problem.

Theorem 5.21. *Let H be a hypergraph based on a weakly connected digraph D, $B \subseteq E(H)$ be a set of size $|V(H)| - 1$, $B_1 := \{e \in B : |e| = 1\}$, and $B_2 := B \setminus B_1$.*

The matrix A_B has rank $|B|$ if and only if

(a) $D[B_1]$ (when seen as a undirected graph) is a forest with $|B_2| + 1$ components, and

(b) if $\{T_r\}_{r\in R}$ is the set of trees of this forest (each rooted at an arbitrary vertex), then for every $r_1, r_2 \in R$ with $r_1 \neq r_2$ the system

$$f(\delta^{in}(r_1)) - f(\delta^{out}(r_1)) = -1 \tag{5.6}$$

$$f(\delta^{in}(r_2)) - f(\delta^{out}(r_2)) = 1 \tag{5.7}$$

$$f(\delta^{in}(v)) - f(\delta^{out}(v)) = 0 \ \forall v \in V(H) \setminus \{r_1, r_2\} \tag{5.8}$$

has a solution $f \in \mathbb{Q}^B$.

Proof. If B is a basis, the linear system $A_B f = d$ has a solution for all $d : V(H) \to \mathbb{Q}$ with $\sum_{v\in V} d(v) = 0$, thus (b) holds. For (a), suppose that $D[B_1]$ contains a cycle C. Orient C in any direction and let $\overrightarrow{C}$ be the arcs of C that agree with the orientation of C and $\overleftarrow{C}$ all other arcs of C. Now,

$$\sum_{e\in\overrightarrow{C}} A_e - \sum_{e\in\overleftarrow{C}} A_e = 0,$$

where A_e denotes the column corresponding to e. In particular, A_B has rank at most $|B| - 1$. Thus, if A_B has rank $|B|$, then $D[B_1]$ is a forest. The number of connected components of $D[B_1]$ is equal to the number of its vertices minus the number of its arcs, which is $|V(H)| - |B_1| = |V(H)| - (|B| - |B_2|) = |B_2| + 1$ as $|B| = |V(H)| - 1$.

Now, suppose B satisfies (a) and (b). The rank of A_B is equal to the dimension of the vector space generated by the columns of A_B. We show that this vector space has dimension $|V(H)| - 1$. A vector $d \in \mathbb{Q}^{V(H)}$ lies in the vector space generated by the columns of A_B if and only if $A_B x = d$ has a solution. We claim that this is the case if and only if $\sum_{v\in V(H)} d(v) = 0$. As $\{d \in \mathbb{Q}^{V(H)} : \sum_{v\in V(H)} d(v) = 0\}$ has dimension $|V(H)| - 1$, this proves our claim.

Let $r_1 \in R$ be a fixed root and denote by f^{r_2} a solution to (5.6)-(5.8) for every $r_2 \in R \setminus \{r_1\}$. For $r \in R$ we set $\delta(r) := \sum_{v\in V(T_r)} d(v)$. We define a vector $y \in \mathbb{Q}^B$ by $y(e) := \sum_{r\in R\setminus\{r_1\}} \delta(r) f^r(e)$ for $e \in B_2$ and $y(e) = 0$ for $e \in B_1$, and a new demand function d' by $d'(v) = d(v) - y(\delta^{in}(v)) + y(\delta^{out}(v))$. We show that d' sums up to zero on each tree T_r. For $r \in R \setminus \{r_1\}$ we get

$$\begin{aligned}\sum_{v\in V(T_r)} d'(v) &= \sum_{v\in V(T_r)} d(v) - \sum_{v\in V(T_r)} \left(y\left(\delta^{in}(v)\right) + y\left(\delta^{out}(v)\right)\right)\\ &= \delta(r) - \sum_{v\in V(T_r)} \left(\sum_{r'\in R\setminus\{r_1\}} \delta(r')\left(f^{r'}\left(\delta^{in}(v)\right) - f^{r'}\left(\delta^{out}(v)\right)\right)\right)\\ &= \delta(r) - \delta(r) = 0,\end{aligned}$$

where we use that $f^{r'}(\delta^{in}(v)) - f^{r'}(\delta^{out}(v)) = 0$ except for $v = r' = r$ for which it is equal to one. For r_1 we get

$$\begin{aligned}\sum_{v\in V(T_{r_1})} d'(v) &= \sum_{v\in V(T_{r_1})} d(v) - \sum_{v\in V(T_{r_1})} \left(y\left(\delta^{in}(v)\right) + y\left(\delta^{out}(v)\right)\right)\\ &= \delta(r_1) - \sum_{v\in V(T_{r_1})} \left(\sum_{r'\in R\setminus\{r_1\}} \delta(r')\left(f^{r'}\left(\delta^{in}(v)\right) - f^{r'}\left(\delta^{out}(v)\right)\right)\right)\\ &= \delta(r_1) + \sum_{r'\in R\setminus\{r_1\}} \delta(r') = \sum_{v\in V(H)} d(v) = 0,\end{aligned}$$

where we use that for every $r' \in R \setminus \{r_1\}$ we have $f^{r'}(\delta^{in}(r_1)) - f^{r'}(\delta^{out}(r_1)) = -1$ and $f^{r'}(\delta^{in}(v)) - f^{r'}(\delta^{out}(v)) = 0$ for $v \in V(T_{r_1}) \setminus \{r_1\}$.

As $\sum_{v\in V(T_r)} d'(v) = 0$, the system $A_{T_r}x = d'_{V(T_r)}$ has a solution, where A_{T_r} denotes the vertex-arc incidence matrix of T_r. For every $r \in R$ let x_r be such a solution. Then the vector $x \in \mathbb{Q}^B$ defined by

$$x(e) := \begin{cases} y(e) & e \in B_2 \\ x_r(e) & e \in E(T_r) \end{cases}$$

solves the system $A_B x = d$. In total, we have shown that $A_B x = d$ has a solution $x \in \mathbb{Q}^B$ for every $d : V(H) \to \mathbb{Q}$ with $\sum_{v\in V(H)} d(v) = 0$, which implies that A_B has rank $|B|$. □

If B is a basis, there even exists a unique solution to (5.6)-(5.8), so the following matrix is well defined.

Definition 5.22. Let H be a hypergraph based on a weakly connected directed graph D, and $B \subseteq E(H)$ be a basis with corresponding rooted trees $\{T_r\}_{r\in R}$. For some fixed root r_1 we define the matrix $TM \in \mathbb{Q}^{B\times R\setminus\{r_1\}}$ as follows:

The column corresponding to $r \in R \setminus \{r_1\}$ contains a solution to (5.6)-(5.8).

We call TM a *treematrix* of H corresponding to the basis B.

We show that every treematrix TM corresponding to a basis B has rank $|B_2|$, and the $|B_2| \times |B_2|$ submatrix of TM consisting of the rows indexed by B_2 is invertible.

Lemma 5.23. *Let H be a hypergraph based on a weakly connected directed graph D, and $B \subseteq E(H)$ be a basis with corresponding rooted trees $\{T_r\}_{r\in R}$.*

If $TM \in \mathbb{Q}^{B\times R\setminus\{r_1\}}$ is a treematrix corresponding to B, then TM has rank $|B_2|$. Moreover, TM restricted to the rows of B_2 is invertible, and its inverse is given by $(|h(e) \cap V(T_r)| - |t(e) \cap V(T_r)|)_{r\in R\setminus\{r_1\}, e\in B_2}$.

Proof. Clearly, TM has rank at most $|R \setminus \{r_1\}| = |B_2|$. Therefore, the second statement implies that TM has rank exactly $|B_2|$.

It is enough to show that $\sum_{e \in B_2} (|h(e) \cap V(T_r)| - |t(e) \cap V(T_r)|)\, TM(e, r')$ is equal to one if $r' = r$ and zero otherwise. For every $r, r' \in R \setminus \{r_1\}$ we have

$$\begin{aligned}&\sum_{v \in V(T_r)} \Big(TM(\delta^{in}(v), r') - TM(\delta^{out}(v), r')\Big)\\ &= \sum_{e \in B} (|h(e) \cap V(T_r)| - |t(e) \cap V(T_r)|) \cdot TM(e, r').\end{aligned}$$

Observe, that $|h(e) \cap V(T_r)| - |t(e) \cap V(T_r) \cap t(e)| = 0$ for all $e \in B_1$ because an arc e lies either completely in $V(T_r)$ or it does not intersect $V(T_r)$. Thus, we get that

$$\begin{aligned}&\sum_{e \in B_2} (|h(e) \cap V(T_r)| - |t(e) \cap V(T_r)|)\, TM(e, r')\\ &= \sum_{v \in V(T_r)} \left(TM\left(\delta^{in}(v), r'\right) - TM\left(\delta^{out}(v), r'\right)\right) = \begin{cases} 1, & r' = r, \\ 0, & r' \in R \setminus \{r_1, r\}. \end{cases} \quad \square\end{aligned}$$

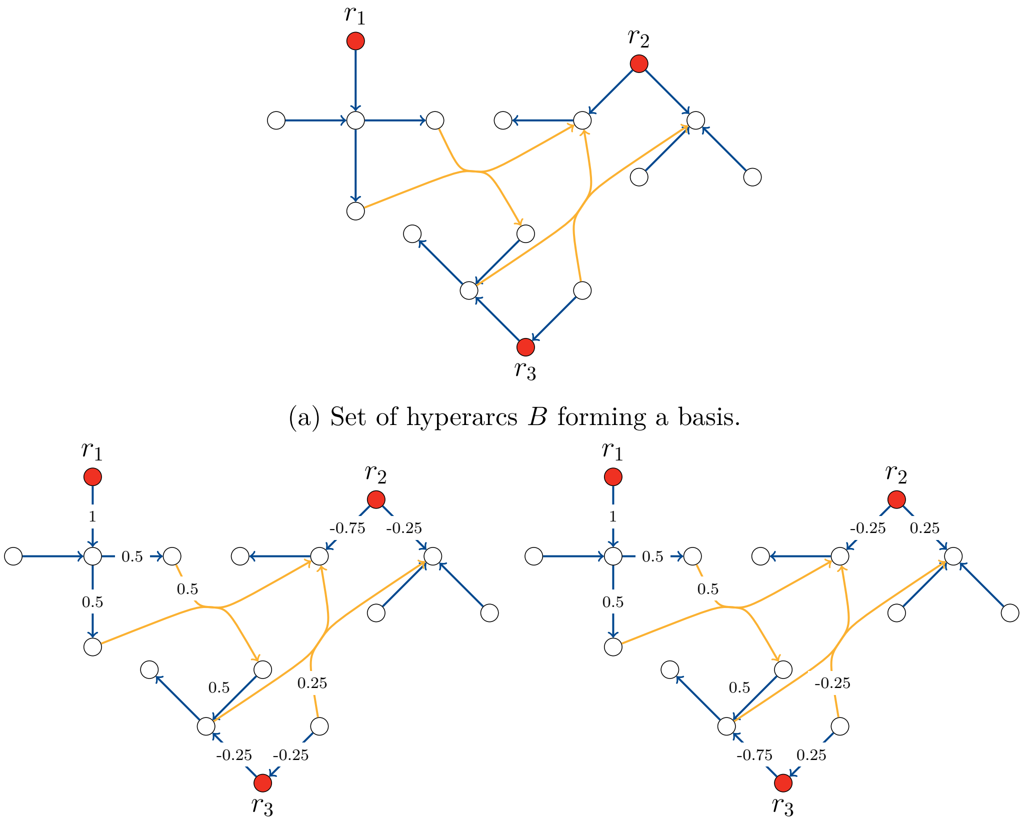
The proof of Theorem 5.21 contains already an algorithm to solve a system of the form $A_B x = d$, where first the flow on the proper hyperarcs is calculated using the matrix TM, and then the flow on the arcs is calculated on each tree separately. For the first part we only need the value of TM at $e \in B_2$. However, we also need the entries at $e \in B_1$ when we "update" the treematrix TM. We do not compute TM from scratch when we change our basis from B to B' during a simplex step, namely, there is a faster way to obtain a treematrix corresponding to B' from one corresponding to B. This is described in the next subsection.

We conclude with an example illustrating Theorem 5.21 and the definition of a treematrix.

Example 5.24. Consider the set of hyperarcs depicted in Figure 5.6a, where we fix root r_1. If we want to send one unit of flow from r_1 to r_2 the hyperarcs have to carry 1/2 and 1/4 unit of flow, and if we send one unit of flow from r_1 to r_3 the hyperarcs carry a flow of 1/2 and $-1/4$, see Figure 5.6. Thus, B is a basis and its treematrix restricted to the set of proper hyperarcs is equal to

$$\begin{pmatrix} 1/2 & 1/2 \\ 1/4 & -1/4 \end{pmatrix},$$

where the first column corresponds to r_2 and the second to r_3.



(a) Set of hyperarcs B forming a basis.

(b) Sending one unit of flow from r_1 to r_2.

(c) Sending one unit of flow from r_1 to r_3.

Figure 5.6: Computation of the treematrix.

5.3.2 A Hypernetwork Simplex Algorithm

Now, we describe a network simplex type algorithm for the minimum cost hyperflow problem on graph-based hypergraphs. Its basic form is the same as in the graph case.

The idea of the network simplex algorithm is to start with a feasible basic solution, then decide whether this solution is optimal, and if not go to a neighboring basic solution. If x^* is a basic solution, then we use linear programming duality to decide whether x^* is of minimum cost. More precisely, we obtain the following optimality condition.

Theorem 5.25. *Let H be a hypergraph based on a digraph D, $d : V(H) \to \mathbb{Q}$ be demands that sum up to zero, and $c : E(H) \to \mathbb{Q}$ be costs on the hyperarcs.*

A feasible basic solution x^* to the minimum cost hyperflow problem with corresponding basis B is of minimum cost with respect to c if and only if there exists a vector $\pi \in \mathbb{Q}^{V(H)}$ such that $c(e) = \sum_{v \in h(e)} \pi(v) - \sum_{v \in t(e)} \pi(v)$ for all $e \in B$, and for every $e \in E(H) \setminus B$ the inequality $c(e) \geq \sum_{v \in h(e)} \pi(v) - \sum_{v \in t(e)} \pi(v)$ holds.

Proof. The theorem follows directly from linear programming duality. However, we give a self-contained proof to give an intuition for the stated optimality condition.

First, let x^* be a basic feasible solution of minimum cost. Let π be a solution to $\pi^T A_B = c_B^T$, which exists as A_B has rank $|B| = |V(H)| - 1$. By the choice of π we have $c(e) = \sum_{v \in h(e)} \pi(v) - \sum_{v \in t(e)} \pi(v)$ for all $e \in B$. Suppose that there exists a hyperarc $e^{in} \in E(H) \setminus B$ with $c(e^{in}) < \sum_{v \in h(e^{in})} \pi(v) - \sum_{v \in t(e^{in})} \pi(v)$. The system $A_B f = -A_{e^{in}}$ has a unique solution f^* because A_B is a basis and $\sum_{v \in V(H)} (-A_{e^{in}})_v = -|h(e^{in})| + |t(e^{in})| = 0$. Let e^{out} be the hyperarc attaining the minimum of $\{x^*(e)/-f^*(e) : f^*(e) < 0, e \in B\}$, and set $\alpha := x^*(e^{out})/-f^*(e^{out})$. We define a new vector $x' \in \mathbb{Q}^{E(H)}$ by

$$x'(e) \leftarrow \begin{cases} \alpha, & e = e^{in} \\ x^*(e) + \alpha \cdot f^*(e), & e \in B \\ x^*(e), & e \in E(H) \setminus (B \cup \{e^{in}\}) \end{cases}, \tag{5.9}$$

and set $B' := B \cup \{e^{in}\} \setminus \{e^{out}\}$. Then, $x'(e) = 0$ for all $e \notin B'$ and $A_{B'} x' = d$. Furthermore, $A_{B'}$ has rank $|B'| = |V(H)| - 1$. Thus, x' is a basic solution, and by the choice of α its entries are non-negative, i.e., it is a feasible basic solution. The cost of x' is

$$\begin{aligned} \sum_{e \in B'} c(e)x'(e) &= \sum_{e \in B} c(e)x^*(e) + \alpha \cdot \sum_{e \in B} c(e) f^*(e) + \alpha \cdot c(e^{in}) \\ &< \sum_{e \in B} c(e)x^*(e) + \alpha \cdot \sum_{e \in B} \left(\sum_{v \in h(e)} \pi(v) - \sum_{v \in t(e)} \pi(v) \right) f^*(e) \\ &\quad + \alpha \left(\sum_{v \in h(e^{in})} \pi(v) - \sum_{v \in t(e^{in})} \pi(v) \right) \\ &= \sum_{e \in B} c(e)x^*(e) + \alpha \cdot \left(\sum_{v \in V(H)} \pi(v) \sum_{e \in B} A_{v,e} f^*(e) \right) \\ &\quad + \alpha \cdot \sum_{v \in V(H)} \pi(v) A_{v,e^{in}} \\ &= \sum_{e \in B} c(e)x^*(e), \end{aligned}$$

where the last equation holds because of $A_B f^* = -A_{e^{in}}$. Thus, x' has smaller cost than x^*, contradicting the optimality of x^*.

On the other hand, let x^* be a basic feasible solution such that a function π with the properties as stated above exists. We have

$$\begin{aligned}\sum_{v \in V(H)} d(v)\pi(v) &= \sum_{v \in V(H)} \pi(v) \cdot \left(\sum_{e \in E(H)} A_{v,e} x^*(e) \right) \\ &= \sum_{v \in V(H)} \pi(v) \cdot \left(\sum_{e \in B} A_{v,e} x^*(e) \right) \\ &= \sum_{e \in B} x^*(e) \left(\sum_{v \in V(H)} A_{v,e} \pi(v) \right) \\ &= \sum_{e \in B} x^*(e) \cdot \left(\sum_{v \in h(e)} \pi(v) - \sum_{v \in t(e)} \pi(v) \right) \\ &= \sum_{e \in B} x^*(e) c(e) = \sum_{e \in E(H)} x^*(e) c(e).\end{aligned}$$

Now, let x' be any other feasible solution to $Ax = d, x \geq 0$, then we obtain

$$\begin{aligned}\sum_{e \in E(H)} c(e) x'(e) &\geq \sum_{e \in E(H)} x'(e) \cdot \left(\sum_{v \in h(e)} \pi(v) - \sum_{v \in t(e)} \pi(v) \right) \\ &= \sum_{v \in V(H)} d(v)\pi(v).\end{aligned}$$

Thus, x^* has minimum cost with respect to c. □

Now, we give a formal description of our network simplex algorithm for the minimum cost hyperflow problem, which we call *hypernetwork simplex algorithm.*

Input: A weakly connected digraph D, a hypergraph H based on D, $d : V(H) \to \mathbb{Q}$ with $\sum_{v \in V(H)} d(v) = 0$, and $c : E(H) \to \mathbb{Q}$.

Output: A minimum cost hyperflow $x : E(H) \to \mathbb{Q}_{\geq 0}$, or "Unbounded".

Initialization: Find a feasible basic flow x on D. Let B a basis corresponding to x, and $T = D[B]$ the spanning tree induced by B, choose a root r_1 arbitrarily, set $T_{r_1} = T$, $R = \{r_1\}$.

1. Solve $\pi^T A_B = c_B^T$ (Dual).

2. Compute the reduced cost $c^\pi(e) = c(e) - \sum_{v \in h(e)} \pi(v) + \sum_{v \in t(e)} \pi(v)$ for all nonbasic hyperarcs $e \in E(H) \backslash B$. If $c^\pi \geq 0$, then output x. Else choose a hyperarc e^{in} with $c^\pi(e^{in}) < 0$ as a hyperarc entering the basis.

3. Solve the system $A_B f = -A_{e^{in}}$ (Primal).

If $f(e) \geq 0$ for all $e \in B$, then output "Unbounded" and stop.

Otherwise, choose a hyperarc e^{out} attaining the minimum of

$\{-x(e)/f(e) : f(e) < 0, e \in B\}$.

4. Set

$$(5.10) \quad x(e) \leftarrow \begin{cases} -x(e^{out})/f(e^{out}), & e = e^{in} \\ x(e) - f(e) \cdot x(e^{out})/f(e^{out}), & e \in B \\ x(e), & e \in E(H) \setminus (B \cup \{e^{in}\}) \end{cases},$$

and $B \leftarrow B \setminus \{e^{out}\} \cup \{e^{in}\}$.
Update the spanning trees T_r, the set of roots R, and the treematrix TM.
Goto 1.

If the hypernetwork simplex algorithm terminates with a flow x, then it returns an optimal minimum cost hyperflow by Theorem 5.25. We can always ensure termination using Bland's pivot rule [Bland, 1977]. That is, we label the hyperarcs from 1 to $|E(H)|$ and if there are several candidates for e^{out} or e^{in} we choose the one with the smallest label. With this pivot rule the simplex method cannot cycle and terminates after a finite number of steps.

Theorem 5.26. *Let H be a hypergraph based on a weakly connected digraph D, $d : V(H) \to \mathbb{Q}$ be demands with $\sum_{v \in V(H)} d(v) = 0$ and $c : E(H) \to \mathbb{Q}$ be costs on the hyperarcs. If e^{in} and e^{out} are chosen according to Bland's rule, then the hypernetwork simplex algorithm will terminate after a finite number of steps. Furthermore, if the minimum cost hyperflow problem on H, d, c is bounded, then the hypernetwork simplex algorithm will return a minimum cost hyperflow, otherwise it returns "Unbounded".*

Proof. By Theorem 1.1 in [Bland, 1977] the hypernetwork simplex algorithm will terminate after a finite number of steps if e^{in} and e^{out} are chosen according to Bland's rule.

As H is based on the digraph D, there exists a feasible hyperflow if and only if there exists a feasible flow on D respecting the demands given by d. As D is weakly connected and the demands sum up to zero, there exists a feasible basic flow x on D. Such a flow corresponds to a spanning tree T of D. Thus, the initialization is always possible.

We claim that during the execution of the algorithm x will always be a feasible basic hyperflow. This is clear at the initialization step. If we update x, then $x(\delta_H(v))$

does not change, and $x(e)$ remains non-negative by the choice of e^{out}. Furthermore, $x(e^{out})$ becomes zero, and $A_{B'}x_{B'} = d$ for $B' = B \setminus \{e^{out}\} \cup \{e^{in}\}$, i.e., x is a feasible basic hyperflow.

We first consider the case that the minimum cost hyperflow problem is bounded. If the hypernetwork simplex algorithm returns a hyperflow x, then x is a minimum cost hyperflow by Theorem 5.25. Otherwise, the algorithm returns "Unbounded". In this case, let x be the hyperflow of the final iteration, e^{in} the chosen entering hyperarc, and f the solution to $A_B f = -A_{e^{in}}$. As the algorithm returns "Unbounded", we know that $f \geq 0$. For any $\alpha \in \mathbb{Z}_{\geq 0}$ the function $x_\alpha : E(H) \to \mathbb{Q}$ defined by

$$x_\alpha(e) \leftarrow \begin{cases} \alpha, & e = e^{in} \\ x(e) + \alpha \cdot f(e), & e \in B \\ x(e), & e \in E(H) \setminus (B \cup \{e^{in}\}) \end{cases}$$

satisfies $x_\alpha(e) \geq 0$, $x_\alpha(\delta_H(v)) = x(\delta_H(v)) = d(v)$ for all $v \in V(H)$, and

$$\begin{aligned} \sum_{e \in E(H)} c(e)x_\alpha(e) &= \sum_{e \in E(H)} c(e)x(e) + \alpha \sum_{e \in E(H)} c(e)f(e) + \alpha f(e^{in}) \\ &= \sum_{e \in E(H)} c(e)x(e) + \alpha c^\pi(e^{in}). \end{aligned}$$

Thus, each x_α is a feasible hyperflow and $c(x_\alpha) \to -\infty$ for $\alpha \to \infty$ as $c^\pi(e^{in}) < 0$. This implies that the hyperflow problem on H, d, c is unbounded, contradicting our assumption.

Now, we consider the case that the hyperflow problem is unbounded. Suppose that the hypernetwork simplex does not return "Unbounded" but a hyperflow x. In this case, let π be the solution to $\pi^T A_B = c_B^T$ computed in the first step of the final iteration. As the algorithm returns x, the reduced cost $c^\pi(e)$ must be non-negative for every $e \in E(H) \setminus B$. Now, let x' be a hyperflow of smaller cost than that of x, which must exists as we assumed our instance to be unbounded. Using $c^\pi \geq 0$ we obtain

$$\begin{aligned} \sum_{e \in E(H)} c(e)x'(e) &\geq \sum_{e \in E(H)} x'(e) \cdot \left(\sum_{v \in h(e)} \pi(v) - \sum_{v \in t(e)} \pi(v) \right) \\ &= \sum_{v \in V(H)} d(v)\pi(v) = \sum_{E \in B} c(e)x(e), \end{aligned}$$

contradicting the choice of x'. Thus, the hypernetwork simplex algorithm correctly returns "Unbounded". □

We have not specified how the systems of linear equations $A_B f = d$ (Primal) and $\pi^T A_B = c_B^T$ (Dual) are solved. We give combinatorial algorithms for both problems.

Given a basis B and corresponding trees $\{T_r\}_{r\in R}$, we always assume that the trees T_r for $r \in R$ have its vertices and arcs ordered such that $v_1 = r$ is the root, v_j is a leaf in $T_r[\{v_1, \ldots, v_j\}]$ and a_{j-1} is the unique arc v_j is incident to in $T_r[\{v_1, \ldots, v_j\}]$. In this way, the arcs of T_r are $a_1, \ldots, a_{|V(T_r)|-1}$ and each a_j is incident to v_{j+1} and some vertex v_i with $i \leq j$. We also assume that $r_1 \in R$ is a fixed root and the treematrix TM is given with respect to this root.

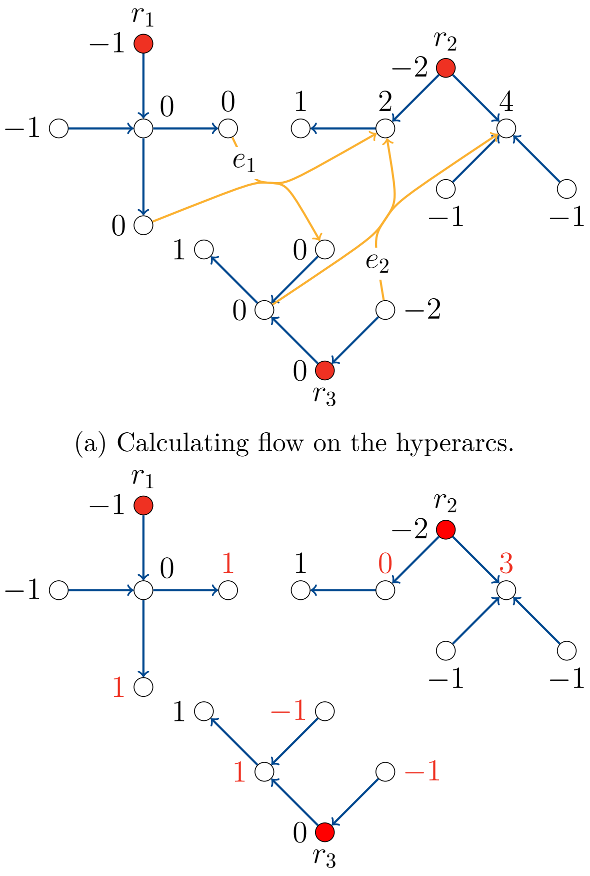
Algorithm 2 Flow

```
procedure FLOW($B$, $\{T_r\}_{r\in R}$, $d$, $f_2$)
    for all $e \in B_2$, $v \in t(e) \cup h(e)$ do
        if $v \in t(e)$ then $d(v) \leftarrow d(v) + f_2(e)$.
        if $v \in h(e)$ then $d(v) \leftarrow d(v) - f_2(e)$.
    end for
    for all trees $T_r$ do
        for $j = |V(T_r)| - 1$ to 1 do
            if $a_j = (v, v_{j+1})$ then $f_1(a_j) \leftarrow d(v_{j+1})$.
            if $a_j = (v_{j+1}, v)$ then $f_1(a_j) \leftarrow -d(v_{j+1})$.
            $d(v) \leftarrow d(v) + d(v_{j+1})$
        end for
    end for
    return $f_1$
end procedure
```

We start with the primal problem $A_B f = d$ for which we basically use the ideas described in the proof of Theorem 5.21. As a subroutine we need Algorithm 2, which given the demand d on the vertices, and flow f_2 on the proper hyperarcs B_2 of the basis B computes the unique flow f_1 on the tree arcs B_1 such that $A_B \cdot \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} = d$, where the columns of A_B are arranged accordingly.

Algorithm 3 Primal

```
procedure PRIMAL
    for all $e \in B_2$ do
        Set $y(e) = \sum_{r\in R\setminus\{r_1\}} TM(e,r) \left(\sum_{v\in V(T_r)} d(v)\right)$.
    end for
    Compute Flow($B$, $\{T_r\}_{r\in R}$, $d$, $y$).
    Set $f(e) = f_1(e)$ for $e \in B_1$, and $f(e) = y(e)$ for $e \in B_2$.
    return $f$
end procedure
```



(a) Calculating flow on the hyperarcs.

(b) Adjusting demands and calculate flow on the arcs.

Figure 5.7: Example of Algorithm 3.

Algorithm 3 describes how $A_B f = d$ is solved. First, we calculate the flow on the proper hyperarcs of the basis using the treematrix TM. Then, we use Algorithm 2 to compute the flow values on the arcs of each tree T_r separately. The correctness of Algorithm 3 directly follows from the second part of the proof of Theorem 5.21 and the definition of the treematrix.

Theorem 5.27. *Algorithm 3 solves correctly the system $A_B f = b$ if A_B is a basis matrix and $\sum_{v \in V} d(v) = 0$.*

Before we show how to solve $\pi^T A_B = c_B^T$ combinatorially, we illustrate Algorithm 3 on an example.

Example 5.28. Let H be a graph based hypergraph with basic arcs and hyperarcs drawn in Figure 5.7a, where the demand of a vertex is given by its label. As seen in Example 5.24, the treematrix of this basis restricted to the rows of B_2 is

$$\begin{pmatrix} 1/2 & 1/2 \\ 1/4 & -1/4 \end{pmatrix}.$$

If we sum up the demands of the vertices in T_{r_2} we obtain 3 and for T_{r_3} we obtain -1. Thus, $f(e_1) = 3/2 - 1/2 = 1$ and $f(e_2) = 3/4 - (-1/4) = 1$. Now, we have to adjust the demands on the vertices $v \in t(e_1) \cup h(e_1) \cup t(e_2) \cup h(e_2)$. The demands of the tail vertices increase by $f(e_1)$ or $f(e_2)$ and that of the head vertices decrease by this amount. The resulting demands are depicted in Figure 5.7b. Finally, we calculate on each tree separately the correct flow values from the leaves to the root. We obtain that $f(e) = 1$ for all arcs e of $T_{r_1}, T_{r_2}, T_{r_3}$.

For the dual problem $\pi^T A_B = c_B^T$ we need Algorithm 4 as a subroutine. Given the cost c_1 of all tree arcs B_1, and the potential π_R at the root vertices the procedure Potential computes a cost vector c_2 on B_2 and potential π_N on the non-root vertices such that $(\pi_R^T, \pi_N^T)A_B = (c_1^T, c_2^T)$, i.e., the reduced cost of every basic arc and hyperarc is zero.

Algorithm 4 Potential

```
procedure POTENTIAL($B, \{T_r\}_{r \in R}, c_1, \pi_R$)
    $\pi'(v) \leftarrow \pi_R(v)$ for all $v \in R$.
    $\pi'(v) \leftarrow 0$ for all $v \in V(H) \setminus R$.
    for all trees $T_r$ do
        for $j = 2$ to $|V(T_r)|$ do
            if $a_{j-1} = (v, v_j)$ then $\pi'(v_j) \leftarrow \pi(v) + c_1(a_j)$.
            if $a_{j-1} = (v_j, v)$ then $\pi'(v_j) \leftarrow \pi(v) - c_1(a_j)$.
        end for
    end for
    for all $e \in B_2$ do
        $c_2(e) \leftarrow \sum_{v \in h(e)} \pi'(v) - \sum_{v \in t(e)} \pi'(v)$.
    end for
    return $c_2, \pi'$.
end procedure
```

As the rank of A_B is $|V(H)| - 1$ the system $\pi^T A_B = c_B^T$ has no unique solution. Thus, we can fix the value of one vertex, for example we can choose $\pi(r_1) = 0$. Now, we can solve $\pi^T A_B = c_B^T$ as shown in Algorithm 5. First, the potential on the root vertices is set to zero, and we compute a potential on the non-root vertices such that the reduced cost of every tree arc is zero. However, some proper hyperarcs might have non-zero reduced costs. Therefore, we calculate potentials on the root vertices using the treematrix TM such that the reduced cost of every proper hyperarc in the basis becomes zero. Finally, the potential on the non-root vertices is adjusted.

Algorithm 5 Dual

```
procedure PRIMAL
    Compute $c_2 \leftarrow$ Potential($B$,$\{T_r\}_{r \in R}$, $c_{|B_1}$, 0).
    Set $y(r_1) = 0$, and $y(r) = \sum_{e \in B_2}(c(e) - c_2(e))TM(e, r)$ for all $r \in R \setminus \{r_1\}$.
    For all $r \in R$ set $\pi(v) \leftarrow \pi'(v) + \pi(r)$ for all $v \in V(T_r)$.
    return $\pi$
end procedure
```

Before we prove that Algorithm 5 works correctly, we illustrate it on the hypergraph from Example 5.28.

Example 5.29. Consider again the basis drawn in Figure 5.7a where the cost of every arc and every hyperarc is one. We want to find $\pi(v)$ for every vertex v such that $\sum_{v \in h(e)} \pi(v) - \sum_{v \in t(e)} \pi(v) = 1$ for all basic arcs and hyperarcs e.

First, we set $\pi(r_1) = \pi(r_2) = \pi(r_3) = 0$ and compute π on each tree $T_{r_1}, T_{r_2}, T_{r_3}$ separately such that $\pi(w) - \pi(v) = 1$ for all arcs (v, w). The resulting potentials are depicted as labels of the vertices in Figure 5.8a.

Using the potentials from the first step we obtain $c_2(e_1) = 1 + 0 - 2 - 2 = -3$ and $c_2(e_2) = 1 + 1 - 1 - (-1) = 2$. Thus, $y(r_2) = 4 \cdot 1/2 + (-1) \cdot 1/4 = 7/4$ and $y(r_3) = 4 \cdot 1/2 + (-1) \cdot (-1/4) = 9/4$. We adjust the potentials by adding $y(r_2)$ to every $\pi(v)$ for $v \in V(T_{r_2})$ and $y(r_3)$ to every $\pi(v)$ for $v \in T_{r_3}$. The resulting potentials are depicted in Figure 5.8b. For every arc (v, w) the potential difference $\pi(w) - \pi(v)$ does not change and $\sum_{v \in h(e_i)} \pi(v) - \sum_{v \in t(e_i)} \pi(v) = 1$ for $i = 1, 2$ after the adjustments of the potentials.

In the following theorem we show that Algorithm 5 not only works correctly on Example 5.29 but also in general.

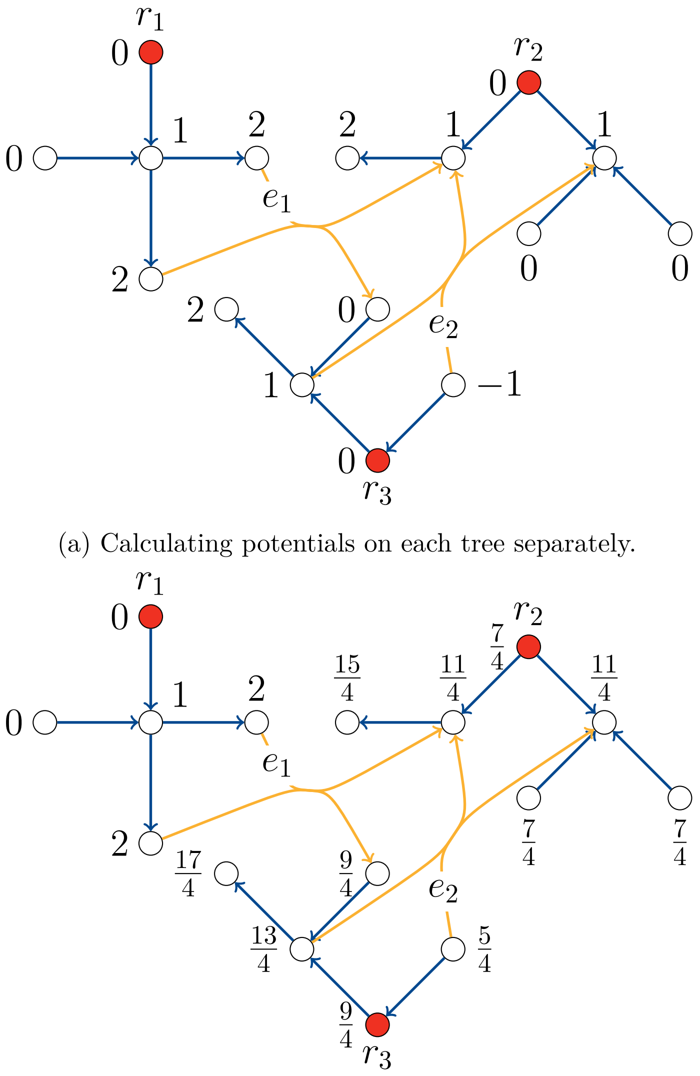
Theorem 5.30. *Algorithm 5 correctly solves the system $\pi^T A_B = c_B^T$ if A_B is a basis matrix.*

Proof. We have to show that the reduced costs $c^\pi(e)$ are zero for all $e \in B$, where $c^\pi(e) = c(e) - \sum_{v \in h(e)} \pi(v) + \sum_{v \in t(e)} \pi(v)$.

For $e \in B_1$ with $e = (v, w)$ and $v, w \in V(T_r)$ we get

$$c^\pi(e) = c(e) - \pi(w) + \pi(v) = c(e) - \pi'(w) - y(r) + \pi'(v) + y(r) = 0.$$

Now, we consider a hyperarc $e' \in B_2$. The matrix $C \in \mathbb{Q}^{(R \setminus \{r_1\}) \times B_2}$ whose entries are $C_{r,e} := |h(e) \cap V(T_r)| - |t(e) \cap V(T_r)|$ is the inverse of the submatrix of TM



(a) Calculating potentials on each tree separately.

(b) Adjusting the potentials.

Figure 5.8: Example of Algorithm 5.

restricted to the rows of B_2, see Lemma 5.23. Therefore, it follows that

$$\begin{aligned}\sum_{r\in R\setminus\{r_1\}} \left(|h(e')\cap V(T_r)| - |t(e')\cap V(T_r)|\right) TM(e,r) &= \sum_{r\in R\setminus\{r_1\}} C_{r,e'}TM(e,r)\\ &= \begin{cases}1, & e=e'\\ 0, & e\neq e'\end{cases}.\end{aligned}$$

Using this equation and $y(r) = \sum_{e\in B_2}(c(e)-c_2(e))TM(e,r)$ for all $r\in R\setminus\{r_1\}$ we obtain

$$\begin{aligned}c^{\pi}(e') &= c(e') - \sum_{v\in h(e')}\pi(v) + \sum_{v\in t(e')}\pi(v)\\ &= c(e') - \sum_{v\in h(e')}\pi'(v) + \sum_{v\in t(e')}\pi'(v)\\ &\quad - \sum_{r\in R\setminus\{r_1\}} \left(|h(e')\cap V(T_r)| - |t(e')\cap V(T_r)|\right) y(r)\\ &= c^{\pi'}(e') - \sum_{r\in R\setminus\{r_1\}} \left(|h(e')\cap V(T_r)| - |t(e')\cap V(T_r)|\right) \sum_{e\in B_2} c^{\pi'}(e)TM(e,r)\\ &= c^{\pi'}(e') - \sum_{e\in B_2} c^{\pi'}(e) \sum_{r\in R\setminus\{r_1\}} \left(|h(e')\cap V(T_r)| - |t(e')\cap V(T_r)|\right) TM(e,r)\\ &= c^{\pi'}(e') - c^{\pi'}(e') = 0.\end{aligned}$$

In total we have shown that $c^{\pi}(e) = 0$ for all $e\in B$, and thus $\pi^T A_B = c_B^T$. □

It remains to show how the treematrix TM is calculated. We do not want to compute TM from scratch for every new basis, instead, we show how to update TM. At the beginning we have a basis consisting only of arcs, and thus TM is the empty matrix. Now, let B be a basis with corresponding treematrix TM, and root set R at the beginning of an iteration of the hypernetwork simplex method. Suppose that $e^{in}\notin B$ enters the basis, and f is the unique solution to $A_B f = -A_{e^{in}}$. If e^{out} is the leaving hyperarc determined by f, then $B' = B\setminus\{e^{out}\}\cup\{e^{in}\}$ a new basis. Let R' be the root set of B' and TM' its treematrix. First, we show that it is possible to choose R' in such a way that it contains at most one vertex not in R and $r_1\in R'$, where r_1 is the root we have chosen at the initialization step.

Observation 5.31. *We can choose R' such that $|R'\setminus R|\leq 1$ and $r_1\in R'$.*

Proof. We prove the observation by distinguishing whether e^{in}, e^{out} are arcs or proper hyperarcs.

1. If e^{out} and e^{in} are both arcs, then the end vertices of e^{in} are either contained in the same tree T_r or one is contained in a tree T_r and the other in a different

tree $T_{r'}$. In the first case, the root set does not change. In the second, e^{out} is an arc of some tree $T_{r''}$, where $r'' = r$ or $r'' = r'$ is possible. We first consider the case that $r'' = r$ or $r'' = r'$. Then, T_r and $T_{r'}$, together with e^{in} and without e^{out} form two new trees T_1, T_2. If $r \in V(T_1)$ and $r' \in V(T_2)$ or the other way around, then $R = R'$. Otherwise, we can assume $r, r' \in V(T_1)$. We have to delete one of r, r' from R and choose a new vertex from $V(T_2)$ that has to be added to R'. Clearly, this can be done such that $r_1 \in R'$ and $|R' \setminus R| = 1$.

Now, we consider the case that $r'' \neq r$ and $r'' \neq r'$. i.e., T_r, $T_{r'}$ and $T_{r''}$ are three distinct trees. Then, T_r, $T_{r'}$ and $T_{r''}$ together with e^{in} and without e^{out} form three new trees T_1, T_2, T_3. One of the three trees, say T_1, is the union of $T_r, T_{r'}$ and e^{in}, the other two are obtained from $T_{r''}$ by deleting e^{out}. One of T_2, T_3 contains r'', for the other one we choose an arbitrary vertex r_3 as its root. We can define R' by adding r_3 to R and deleting one of r, r' in such a way that $r_1 \in R'$.

2. Now, assume e^{out} is an arc and e^{in} a proper hyperarc. The forest corresponding to B' has one more tree than that corresponding to B. Namely, the tree T_r containing e^{out} is split into two trees. For the one not containing r, we have to choose a vertex r' as a new root. Again, $r_1 \in R'$ and $|R' \setminus R| = 1$ holds.

3. If e^{out} is a proper hyperarc and e^{in} an arc, then e^{in} connects two trees T_r, $T_{r'}$ that become one tree in B'. To obtain R' from R we have to delete one of r, r'. If one of the two vertices is r_1, we delete the other vertex, otherwise, we delete just any of the two. By this way we get $r_1 \in R'$ and $|R' \setminus R| = 0$.

4. Finally, if e^{out} and e^{in} are both proper hyperarcs, then the trees $\{T_r\}_{r \in R}$ do not change and we can set $R' = R$. □

In order to update TM, we define a function $\tilde{f} : (B \cup \{e^{in}\}) \to \mathbb{Z}$ by $\tilde{f}(e) = f(e)$ for all $e \in B$, and $\tilde{f}(e^{in}) = 1$, where f is the solution to $A_B f = -A_{e^{in}}$ computed in the third step of the current iteration of the hypernetwork simplex algorithm. Furthermore, for every $r \in (R' \cap R) \setminus \{r_1\}$ we define a function $f^r : (B \cup \{e^{in}\}) \to \mathbb{Q}$ by $f^r(e) := TM(e, r)$ for all $e \in B$ and $f^r(e^{in}) = 0$. If there exists a new root $r \in R' \setminus R$ we calculate a solution f^r to $A|_{(B \cup \{e^{in}\})} x = \delta_r$ with $f^r(e^{in}) = 0$, where

$$\delta_r(v) := \begin{cases} 1, & \text{if } v = r \\ -1, & \text{if } v = r_1 \\ 0, & \text{otherwise} \end{cases} .$$

This can be done via Algorithm 3 (using the treematrix TM of basis B). Finally, for every $r \in R' \setminus \{r_1\}$ we set $\alpha_r := \frac{f^r(e^{out})}{\tilde{f}(e^{out})}$. Now, we update TM as follows:

For all $e \in B'$ and $r \in R' \setminus \{r_1\}$ we set $TM'(e,r) = f^r(e) - \alpha_r \cdot \tilde{f}(e)$.

It remains to show that the treematrix is updated correctly. Therefore, we have to prove that $TM'(\cdot, r)$ satisfies (5.6)-(5.8) for all $r \in R' \setminus \{r_1\}$. First, we observe that $f^r(e^{out}) - \alpha \cdot \tilde{f}(e^{out}) = 0$. Thus, $f^r(e) - \alpha \cdot \tilde{f}(e) \neq 0$ is only possible for $e \in B'$. For any $r \in R' \setminus \{r_1\}$ and $v \in V(H)$ we have

$$\begin{aligned}
&\sum_{e \in \delta^{in}(v)} TM'(e,r) - \sum_{e \in \delta^{out}(v)} TM'(e,r) \\
&= f^r(\delta^{in}(v)) - f^r(\delta^{out}(v)) - \alpha \cdot \left(\tilde{f}(\delta^{in}(v)) - \tilde{f}(\delta^{out}(v))\right) \\
&= f^r(\delta^{in}(v)) - f^r(\delta^{out}(v)).
\end{aligned}$$

This calculation implies that each column of TM' satisfies (5.6)-(5.8).

Although we only need the entries $TM(e,r)$ of the treematrix for $e \in B_2$ during the execution of Algorithm 3 and Algorithm 5, we also need the entries at $e \in B_1$ during the update of the treematrix because it is possible that $e^{out} \in B_1$. If we only save $TM(e,r)$ for $e \in B_2$, then we have to calculate $f^r(e^{out})$ for all $R' \setminus \{r_1\}$ if $e^{out} \in B_1$, for which we would need $|R'| - 1 = |B'_2|$ calls to Algorithm 3. This would lead to a worse running time for the hypernetwork simplex method. Thus, we save $TM(e,r)$ for all $e \in B$.

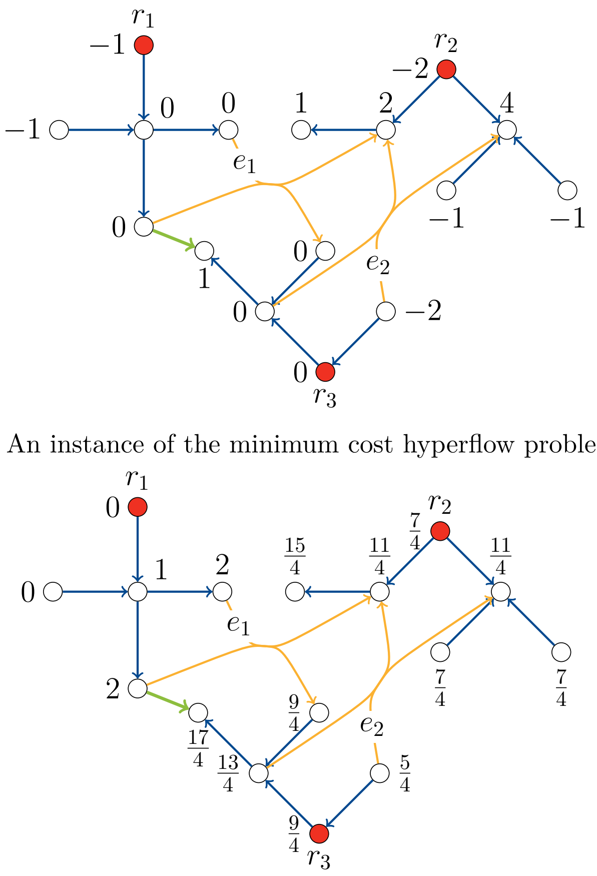
Regarding the time complexity we get the following result.

Lemma 5.32. *Step 1. to 4. in the hypernetwork simplex method can be done in $\mathcal{O}(|V(H)| \cdot |E(H)|)$-time.*

Proof. The complexity of Algorithm 2 and 4 is $\mathcal{O}(|B_2| \cdot |V(H)|)$. Thus, Algorithm 3 and 5 can be done in time $\mathcal{O}(|R| \cdot |B_2| + |B_2| \cdot |V(H)|) = \mathcal{O}(|V(H)|^2)$. Computing the reduced cost of all non-basic hyperarcs takes time $\mathcal{O}(|V(H)| \cdot |E(H)|)$. As we call Algorithm 5 once and Algorithm 3 at most twice during steps 1. to 4., the total time complexity is $\mathcal{O}(|V(H)| \cdot |E(H)|)$. □

In general, a graph-based hypergraph will have much more hyperarcs than vertices. Thus, computing the reduced costs will be the most time consuming step during the simplex method.

Though each of the steps in the presented hypernetwork simplex method can be done in polynomial time, it is not a polynomial time algorithm. The reason is that it might use an exponential number of iterations. In particular, Bland's rule uses in practice often much more pivot steps than other pivoting rules. For the network simplex method there exists another trick to avoid cycling. Namely, one only considers bases corresponding to special spanning trees, called strongly feasible spanning trees. It can be shown that if one starts with a strongly feasible spanning tree, then one can always update the spanning tree such that it remains strongly feasible. Working with strongly feasible trees guarantees that the network simplex algorithm terminates after a finite number of iterations independent of the chosen pivot rule, for more details see [Cunningham, 1976].



(a) An instance of the minimum cost hyperflow problem.

(b) Potentials such that all basic hyperarcs have zero reduced cost.

Figure 5.9: Example for the hypernetwork flow algorithm.

The use of strongly feasible trees in the network simplex algorithm is important both in theory and in practice. Namely, Orlin's polynomial time variant of the network simplex algorithm in [Orlin, 1997] uses them, and in practice it turns out that the chosen pivot rule has a huge effect on the performance of the network simplex algorithm, see for example [Löbel, 1996]. It does not seem obvious how to generalize the idea of strongly feasible trees to the hypergraph case. At least one can guarantee finite termination by choosing a simplex rule that does not cycle.

We conclude this chapter by illustrating one iteration of the hypernetwork simplex method using our running example.

Example 5.33. Consider an instance of the minimum cost hyperflow problem on the hypergraph shown in Figure 5.9a in which all hyperarcs and arcs have cost one, and the demands are given as labels on the vertices. The arcs and proper hyperarcs in the basis are drawn in blue and orange, and the unique non-basic arc is drawn in green. The current solution corresponding to this basis is $x(e) = 1$ for all basic hyperarcs e and $x(e) = 0$ otherwise.

In the first step of the hypernetwork simplex algorithm we have to solve the linear system $\pi^T A_B = c_B^T$ using Algorithm 5. This was done in Example 5.29 and the result is shown in Figure 5.9b. The reduced cost of the green arc is $1 - 17/4 + 2 = -5/4$, which is negative. Thus, we choose this arc as the arc e^{in} entering the basis.

In Step 3 of the hypernetwork simplex algorithm we have to find e^{out} such that $B \setminus \{e^{out}\} \cup \{e^{in}\}$ is again a basis. Therefore, we have to solve $A_B f = -A_{e^{in}}$ using Algorithm 3. If we denote by w the tail and by u the head of the green arc, then w gets a demand of 1, u of -1, and all other vertices have zero demand, see Figure 5.11a. Figure 5.11b shows the solution of $A_B f = -A_{e^{in}}$ returned by Algorithm 3.

Finally, we choose a basic hyperarc e^{out} attaining the minimum of $x(e)/-f(e)$ over all e with $f(e) < 0$. In this case, the arc drawn in red in Figure 5.11c is the unique hyperarc minimizing $x(e)/-f(e)$. Augmenting x by $1 = x(e^{out})/-f(e^{out})$ units of flow along f gives a new basic solution to the minimum cost hyperflow problem. The new flow values are depicted in Figure 5.10.

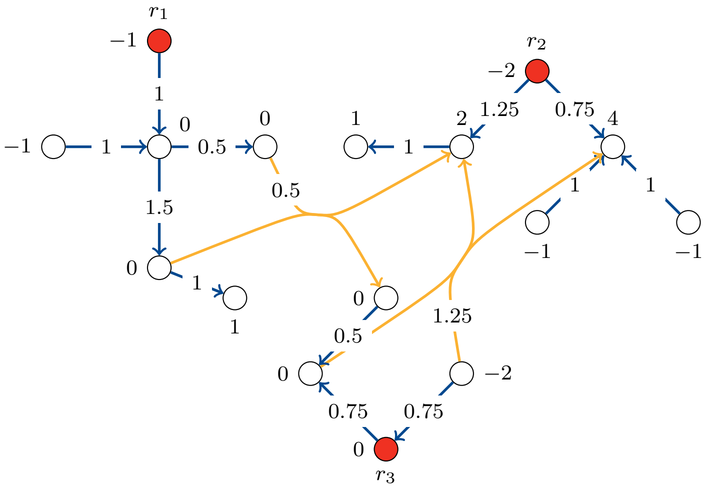
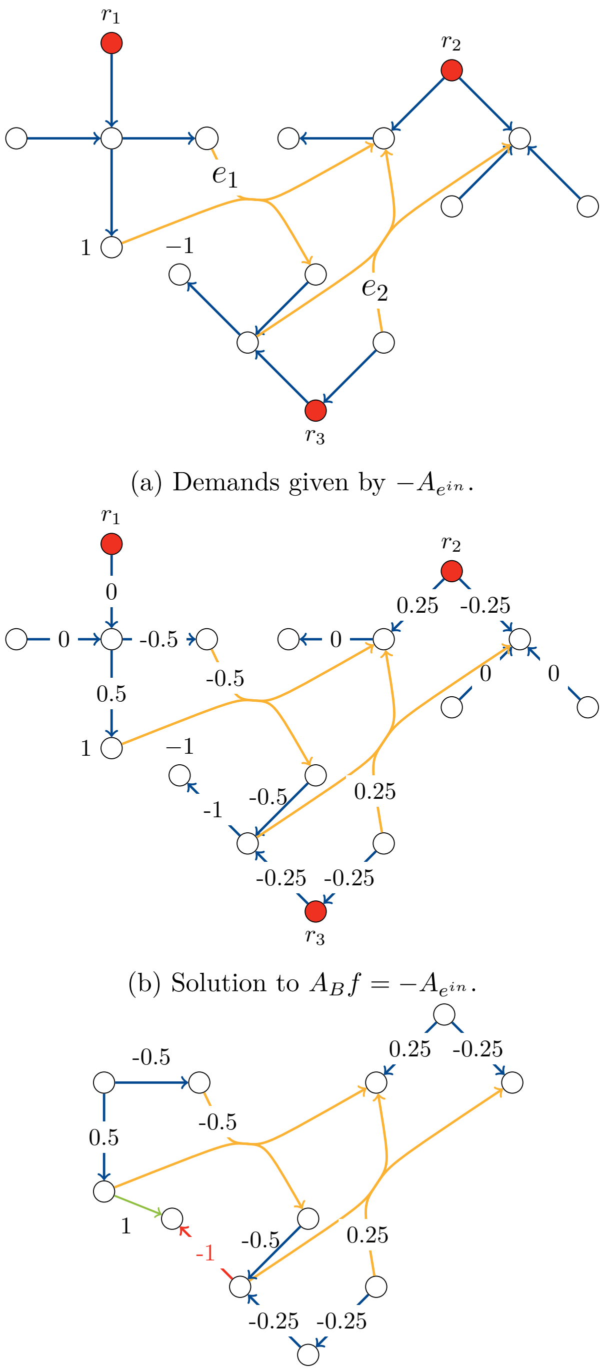


Figure 5.10: Update of the basis and the flow function.



(a) Demands given by $-A_{e^{in}}$.

(b) Solution to $A_B f = -A_{e^{in}}$.

(c) Determining leaving arc e^{out}.

Figure 5.11: Step 3 of the hypernetwork simplex algorithm.

A Zusammenfassung

Diese Arbeit untersucht Matchings und Flüsse in Hypergraphen mit Hilfe kombinatorischer Methoden. In Graphen gehören diese Probleme zu den grundlegendsten der kombinatorischen Optimierung. Viele Resultate lassen sich nicht von Graphen auf Hypergraphen verallgemeinern, da Hypergraphen ein sehr abstraktes Konzept bilden. Daher schauen wir uns bestimmte Klassen von Hypergraphen an, die mehr Struktur besitzen, und nutzen diese aus um Resultate aus der Graphentheorie zu übertragen.

In Kapitel 2 betrachten wir das perfekte Matchingproblem auf Klassen von „bipartiten“ Hypergraphen, wobei es verschiedene Möglichkeiten gibt den Begriff „bipartit“ auf Hypergraphen zu definieren. Für sogenannte partitionierte Hypergraphen geben wir einen polynomiellen Approximationsalgorithmus an, dessen Gütegarantie bis auf eine Konstante bestmöglich ist. Danach betrachten wir die Sätze von Kőnig und Hall und untersuchen deren Zusammenhang. Unser Hauptresultat ist eine Bedingung für die Existenz von perfekten Matchings auf normalen Hypergraphen, die Halls Bedingung für bipartite Graphen verallgemeinert.

Als Verallgemeinerung von perfekten Matchings betrachten wir in Kapitel 3 perfekte f-Matchings, f-Faktoren und (g, f)-Matchings. Wir beweisen Bedingungen für die Existenz von (g, f)-Matchings auf unimodularen Hypergraphen, perfekten f-Matchings auf uniformen Mengerschen Hypergraphen und f-Faktoren auf uniformen balancierten Hypergraphen. Außerdem geben wir eine Übersicht über die Komplexität des (g, f)-Matchingproblems auf verschiedenen Klassen von Hypergraphen an, die bipartite Graphen verallgemeinern.

In Kapitel 4 untersuchen wir die Struktur von Hypergraphen, die ein perfektes Matching besitzen. Wir zeigen, dass diese Hypergraphen entlang spezieller Schnitte zerlegt werden können. Für Graphen weiß man, dass die so erhaltene Zerlegung eindeutig ist, was im Allgemeinen für Hypergraphen nicht zutrifft. Wenn man jedoch uniforme Hypergraphen betrachtet, dann liefert jede Zerlegung die gleichen unzerlegbaren Hypergraphen bis auf parallele Hyperkanten.

Kapitel 5 beschäftigt sich mit Flüssen in gerichteten Hypergraphen, wobei wir Hypergraphen betrachten, die auf gerichteten Graphen basieren. Das bedeutet, dass eine Hyperkante die Vereinigung einer Menge von disjunkten Kanten ist. Wir definieren ein Residualnetzwerk, mit dessen Hilfe man entscheiden kann, ob ein gegebener Fluss optimal ist. Unser Hauptresultat in diesem Kapitel ist ein Algorithmus, um einen Fluss minimaler Kosten zu finden, der den Netzwerksimplex verallgemeinert.

Bibliography

[Aharoni, 2001] Aharoni, R. (2001). Ryser's Conjecture for Tripartite 3-Graphs. *Combinatorica*, 21(1):1–4.

[Aharoni et al., 2009] Aharoni, R., Georgakopoulos, A., and Sprüssel, P. (2009). Perfect Matchings in r-Partite r-Graphs. *European Journal of Combinatorics*, 30(1):39–42.

[Aharoni and Haxell, 2000] Aharoni, R. and Haxell, P. (2000). Hall's Theorem for Hypergraphs. *Journal of Graph Theory*, 35(2):83–88.

[Aharoni and Kessler, 1990] Aharoni, R. and Kessler, O. (1990). On a Possible Extension of Hall's Theorem to Bipartite Hypergraphs. *Discrete Mathematics*, 84(3):309–313.

[Ahuja et al., 1993] Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. (1993). *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall.

[Aissi et al., 2014] Aissi, H., Mahjoub, A. R., McCormick, S. T., and Queyranne, M. (2014). A Strongly Polynomial Time Algorithm for Multicriteria Global Minimum Cuts. In *International Conference on Integer Programming and Combinatorial Optimization*, pages 25–36. Springer.

[Akiyama and Kano, 2011] Akiyama, J. and Kano, M. (2011). *Factors and Factorizations of Graphs: Proof Techniques in Factor Theory*, volume 2031. Springer.

[Anbeek et al., 1997] Anbeek, C., DeTemple, D., McAvaney, K., and Robertson, J. (1997). When are Chordal Graphs also Partition Graphs? *Australasian Journal of Combinatorics*, 16:285–293.

[Annamalai, 2016] Annamalai, C. (2016). Finding Perfect Matchings in Bipartite Hypergraphs. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1814–1823. Society for Industrial and Applied Mathematics.

[Anstee, 1985] Anstee, R. (1985). An Algorithmic Proof of Tutte's f-Factor Theorem. *Journal of Algorithms*, 6(1):112–131.

[Artmann et al., 2017] Artmann, S., Weismantel, R., and Zenklusen, R. (2017). A Strongly Polynomial Algorithm for Bimodular Integer Linear Programming. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2017, pages 1206–1219, New York, NY, USA. ACM.

[Ausiello et al., 2001] Ausiello, G., Franciosa, P. G., and Frigioni, D. (2001). Directed Hypergraphs: Problems, Algorithmic Results, and a Novel Decremental Approach. In *Italian Conference on Theoretical Computer Science*, pages 312–328. Springer.

[Ausiello and Laura, 2017] Ausiello, G. and Laura, L. (2017). Directed Hypergraphs: Introduction and Fundamental Algorithms—A Survey. *Theoretical Computer Science*, 658, Part B:293–306.

[Beckenbach, 2018] Beckenbach, I. (2018). A Hypergraph Network Simplex Algorithm. In *Operations Research Proceedings 2017*, pages 309–315. Springer International Publishing.

[Beckenbach and Borndörfer, 2016] Beckenbach, I. and Borndörfer, R. (2016). An Approximation Result for Matchings in Partitioned Hypergraphs. In *Operations Research Proceedings 2014*, pages 31–36. Springer International Publishing.

[Beckenbach and Borndörfer, 2018] Beckenbach, I. and Borndörfer, R. (2018). Hall's and Kőnig's Theorem in Graphs and Hypergraphs. *Discrete Mathematics*, 341(10):2753–2761.

[Beckenbach et al., 2017] Beckenbach, I., Peis, B., Schaudt, O., and Scheidweiler, R. (2017). A Combinatorial Proof of a Kőnig-type Theorem for Unimodular Hypergraphs. Technical Report 17-27, ZIB, Takustr. 7, 14195 Berlin.

[Beckenbach and Scheidweiler, 2017] Beckenbach, I. and Scheidweiler, R. (2017). Perfect f-Matchings and f-Factors in Hypergraphs—A Combinatorial Approach. *Discrete Mathematics*, 340(10):2499–2506.

[Berge, 1970] Berge, C. (1970). Sur Certains Hypergraphes Généralisant les Graphes Bipartites. *Combinatorial Theory and its Applications*, 4:119–133.

[Berge, 1975] Berge, C. (1975). Isomorphism Problems for Hypergraphs. In Hall, M. and van Lint, J. H., editors, *Combinatorics*, pages 205–214. Springer Netherlands.

[Berge, 1984] Berge, C. (1984). *Hypergraphs: Combinatorics of Finite Sets*, volume 45. Elsevier.

[Berge and Las Vergnas, 1970] Berge, C. and Las Vergnas, M. (1970). Sur un Theorem du Type Kőnig pour Hypergraphes. *Annals of the New York Academy of Sciences*, 175(1):32–40.

[Berman, 2000] Berman, P. (2000). A d/2 Approximation for Maximum Weight Independent Set in d-Claw Free Graphs. In *Scandinavian Workshop on Algorithm Theory*, pages 214–219. Springer.

[Bland, 1977] Bland, R. G. (1977). New Finite Pivoting Rules for the Simplex Method. *Mathematics of Operations Research*, 2(2):103–107.

[Borndörfer, 1998] Borndörfer, R. (1998). *Aspects of Set Packing, Partitioning, and Covering*. PhD thesis, TU Berlin.

[Borndörfer and Heismann, 2012] Borndörfer, R. and Heismann, O. (2012). Minimum Cost Hyperassignments with Applications to ICE/IC Rotation Planning. In *Operations Research Proceedings 2011*, pages 59 – 64. Springer International Publishing.

[Borndörfer et al., 2011] Borndörfer, R., Reuther, M., Schlechte, T., and Weider, S. (2011). A Hypergraph Model for Railway Vehicle Rotation Planning. In *OASIcs-OpenAccess Series in Informatics*, volume 20. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

[Borndörfer et al., 2012] Borndörfer, R., Reuther, M., Schlechte, T., and Weider, S. (2012). Vehicle Rotation Planning for Intercity Railways. *Proceedings of Conference on Advanced Systems for Public Transport 2012 (CASPT12).*

[Boros et al., 2017] Boros, E., Gurvich, V., and Milanič, M. (2017). On Equistable, Split, CIS, and Related Classes of Graphs. *Discrete Applied Mathematics*, 216:47–66.

[Bose, 1938] Bose, R. C. (1938). On the Application of the Properties of Galois Fields to the Problem of Construction of Hyper-Græco-Latin Squares. *Sankhyā: The Indian Journal of Statistics (1933-1960)*, 3(4):323–338.

[Brandstädt et al., 1999] Brandstädt, A., Le, V. B., and Spinrad, J. P. (1999). *Graph Classes: A Survey.* SIAM monographs on discrete mathematics and applications. SIAM.

[Cambini et al., 1997] Cambini, R., Gallo, G., and Scutellà, M. G. (1997). Flows on Hypergraphs. *Mathematical Programming*, 78(2):195–217.

[Cambini et al., 1992] Cambini, R., Gallo, G., Scutellà, M. G., et al. (1992). Minimum Cost Flows on Hypergraphs. Technical report, Dipartimento di Informatica, Università di Pisa.

[Caro et al., 1996] Caro, Y., Sebő, A., and Tarsi, M. (1996). Recognizing Greedy Structures. *Journal of Algorithms*, 20(1):137–156.

[Chan and Lau, 2012] Chan, Y. H. and Lau, L. C. (2012). On Linear and Semidefinite Programming Relaxations for Hypergraph Matching. *Mathematical Programming*, 135(1):123–148.

[Chandrasekaran et al., 2018] Chandrasekaran, K., Xu, C., and Yu, X. (2018). Hypergraph k-Cut in Randomized Polynomial Time. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '18, pages 1426–1438, Philadelphia, PA, USA. Society for Industrial and Applied Mathematics.

[Chekuri and Xu, 2017] Chekuri, C. and Xu, C. (2017). Computing Minimum Cuts in Hypergraphs. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '17, pages 1085–1100, Philadelphia, PA, USA. Society for Industrial and Applied Mathematics.

[Chvátal and Slater, 1993] Chvátal, V. and Slater, P. J. (1993). A Note on Well-Covered Graphs. In Gimbel, J., Kennedy, J. W., and Quintas, L. V., editors, *Quo Vadis, Graph Theory?*, volume 55 of *Annals of Discrete Mathematics*, pages 179–181. Elsevier.

[Conforti and Cornuéjols, 1987] Conforti, M. and Cornuéjols, G. (1987). An Algorithmic Framework for the Matching Problem in Some Hypergraphs. *Networks*, 17(4):365–386.

[Conforti et al., 1996] Conforti, M., Cornuéjols, G., Kapoor, A., and Vušković, K. (1996). Perfect Matchings in Balanced Hypergraphs. *Combinatorica*, 16(3):325–329.

[Conforti et al., 2001a] Conforti, M., Cornuéjols, G., Kapoor, A., and Vušković, K. (2001a). Balanced 0,±1 Matrices I. Decomposition. *Journal of Combinatorial Theory, Series B*, 81(2):243–274.

[Conforti et al., 2001b] Conforti, M., Cornuéjols, G., Kapoor, A., and Vušković, K. (2001b). Balanced 0,±1 Matrices II. Recognition Algorithm. *Journal of Combinatorial Theory, Series B*, 81(2):275–306.

[Conforti et al., 1999] Conforti, M., Cornuéjols, G., and Rao, M. (1999). Decomposition of Balanced Matrices. *Journal of Combinatorial Theory, Series B*, 77(2):292–406.

[Conforti et al., 2006] Conforti, M., Cornuéjols, G., and Vušković, K. (2006). Balanced Matrices. *Discrete Mathematics*, 306(19):2411–2437.

[Cornuéjols, 2001] Cornuéjols, G. (2001). *Combinatorial Optimization: Packing and Covering*. SIAM.

[Crama et al., 1990] Crama, Y., Hammer, P. L., and Ibaraki, T. (1990). Packing, Covering and Partitioning Problems with Strongly Unimodular Constraint Matrices. *Mathematics of Operations Research*, 15(2):258–267.

[Cunningham, 1976] Cunningham, W. H. (1976). A Network Simplex Method. *Mathematical Programming*, 11(1):105–116.

[Cunningham, 1983] Cunningham, W. H. (1983). Decomposition of Submodular Functions. *Combinatorica*, 3(1):53–68.

[Cygan, 2013] Cygan, M. (2013). Improved Approximation for 3-Dimensional Matching via Bounded Pathwidth Local Search. In *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*, pages 509–518. IEEE.

[de Carvalho et al., 2002] de Carvalho, M. H., Lucchesi, C. L., and Murty, U. (2002). On a Conjecture of Lovász Concerning Bricks: I. The Characteristic of a Matching Covered Graph. *Journal of Combinatorial Theory, Series B*, 85(1):94–136.

[de Carvalho et al., 2004] de Carvalho, M. H., Lucchesi, C. L., and Murty, U. (2004). The Perfect Matching Polytope and Solid Bricks. *Journal of Combinatorial Theory, Series B*, 92(2):319–324.

[de Carvalho et al., 2006] de Carvalho, M. H., Lucchesi, C. L., and Murty, U. (2006). How to Build a Brick. *Discrete Mathematics*, 306(19):2383–2410.

[de Werra, 1971] de Werra, D. (1971). Equitable Colorations of Graphs. *Revue française d'informatique et de recherche opérationnelle*, pages 3–8.

[Dean and Zito, 1994] Dean, N. and Zito, J. (1994). Well-Covered Graphs and Extendability. *Discrete Mathematics*, 126(1):67 – 80.

[Deming, 1979] Deming, R. W. (1979). Independence Numbers of Graphs-An Extension of the Kőnig-Egerváry Theorem. *Discrete Mathematics*, 27(1):23–33.

[Deng et al., 1999] Deng, X., Ibaraki, T., and Nagamochi, H. (1999). Algorithmic Aspects of the Core of Combinatorial Optimization Games. *Mathematics of Operations Research*, 24(3):751–766.

[DeTemple et al., 1987] DeTemple, D., Harary, F., and Robertson, J. (1987). Partition Graphs. *Soochow Journal of Mathematics*, 13(2):121–129.

[Edmonds, 1965a] Edmonds, J. (1965a). Maximum Matching and a Polyhedron with 0,1-Vertices. *Journal of Research of the National Bureau of Standards B*, 69(125-130):55–56.

[Edmonds, 1965b] Edmonds, J. (1965b). Paths, Trees, and Flowers. *Canadian Journal of Mathematics*, 17(3):449–467.

[Edmonds et al., 1982] Edmonds, J., Pulleyblank, W. R., and Lovász, L. (1982). Brick Decompositions and the Matching Rank of Graphs. *Combinatorica*, 2(3):247–274.

[Eiter and Gottlob, 1995] Eiter, T. and Gottlob, G. (1995). Identifying the Minimal Transversals of a Hypergraph and Related Problems. *SIAM Journal on Computing*, 24(6):1278–1304.

[Ford and Fulkerson, 1956] Ford, L. R. and Fulkerson, D. R. (1956). Maximal Flow through a Network. *Canadian journal of Mathematics*, 8(3):399–404.

[Frank, 2011] Frank, A. (2011). *Connections in Combinatorial Optimization.* Oxford University Press, Oxford.

[Fulkerson et al., 1965] Fulkerson, D. R., Hoffman, A. J., and McAndrew, M. H. (1965). Some Properties of Graphs with Multiple Edges. *Canadian Journal of Mathematics*, 17:166–177.

[Fulkerson et al., 1974] Fulkerson, D. R., Hoffman, A. J., and Oppenheim, R. (1974). On Balanced Matrices. *Pivoting and Extension*, pages 120–132.

[Füredi, 1981] Füredi, Z. (1981). Maximum Degree and Fractional Matchings in Uniform Hypergraphs. *Combinatorica*, 1(2):155–162.

[Füredi et al., 1993] Füredi, Z., Kahn, J., and Seymour, P. D. (1993). On the Fractional Matching Polytope of a Hypergraph. *Combinatorica*, 13(2):167–180.

[Fürer and Yu, 2014] Fürer, M. and Yu, H. (2014). Approximating the k-Set Packing Problem by Local Improvements. In *Combinatorial Optimization: Third International Symposium, ISCO 2014, Lisbon, Portugal, March 5-7, 2014, Revised Selected Papers*, pages 408–420. Springer International Publishing.

[Gallo et al., 1993] Gallo, G., Longo, G., Pallottino, S., and Nguyen, S. (1993). Directed Hypergraphs and Applications. *Discrete Applied Mathematics*, 42(2):177–201.

[Ghouila-Houri, 1962] Ghouila-Houri, A. (1962). Caractérisation des Matrices Totalement Unimodulaires. *Comptes Rendus Hebdomadaires des Séances de l'Académie des Science*, 254:1192–1193.

[Goldschmidt and Hochbaum, 1994] Goldschmidt, O. and Hochbaum, D. S. (1994). A Polynomial Algorithm for the k-Cut Problem for Fixed k. *Mathematics of Operations Research*, 19(1):24–37.

[Grötschel et al., 1988] Grötschel, M., Lovász, L., and Schrijver, A. (1988). *Geometric Algorithms and Combinatorial Optimization*, volume 2 of *Algorithms and Combinatorics*. Springer.

[Hackfeld and Koster, 2018] Hackfeld, J. and Koster, A. (2018). The Matching Extension Problem in General Graphs is co-NP-Complete. *Journal of Combinatorial Optimization*, 35(3):853–859.

[Halldórsson, 1999] Halldórsson, M. M. (1999). Approximations of Weighted Independent Set and Hereditary Subset Problems. In *Computing and Combinatorics: 5th Annual International Conference, COCOON'99 Tokyo, Japan, July 26–28, 1999 Proceedings*, pages 261–270. Springer Berlin Heidelberg.

[Halldórsson et al., 2000] Halldórsson, M. M., Kratochvíl, J., and Telle, J. A. (2000). Independent Sets with Domination Constraints. *Discrete Applied Mathematics*, 99(1–3):39–54.

[Haxell, 1995] Haxell, P. E. (1995). A Condition for Matchability in Hypergraphs. *Graphs and Combinatorics*, 11(3):245–248.

[Hazan et al., 2006] Hazan, E., Safra, S., and Schwartz, O. (2006). On the Complexity of Approximating k-Set Packing. *Computational Complexity*, 15(1):20–39.

[Heinrich et al., 1990] Heinrich, K., Hell, P., Kirkpatrick, D. G., and Liu, G. (1990). A Simple Existence Criterion for (g < f)-Factors. *Discrete Mathematics*, 85(3):313–317.

[Henderson, 1971] Henderson, J. R. (1971). *Permutation Decomposition of (0,1)-Matrices and Decomposition Transversals*. PhD thesis, California Institute of Technology. `https://thesis.library.caltech.edu/5726/`.

[Hoffman, 1960] Hoffman, A. J. (1960). Some Recent Applications of the Theory of Linear Inequalities to Extremal Combinatorial Analysis. In *Proceedings of Symposia in Applied Mathematics*, volume 10, pages 113–127. World Scientific.

[Hoffman, 1974] Hoffman, A. J. (1974). A Generalization of Max Flow-Min Cut. *Mathematical Programming*, 6(1):352–359.

[Hoffman, 1976] Hoffman, A. J. (1976). Total Unimodularity and Combinatorial Theorems. *Linear Algebra and its Applications*, 13(1):103–108.

[Hoffman and Kruskal, 2010] Hoffman, A. J. and Kruskal, J. B. (2010). Integral Boundary Points of Convex Polyhedra. In *50 Years of Integer Programming 1958-2008: From the Early Years to the State-of-the-Art*, pages 49–76. Springer Berlin Heidelberg, Berlin, Heidelberg.

[Huck and Triesch, 2002] Huck, A. and Triesch, E. (2002). Perfect Matchings in Balanced Hypergraphs–A Combinatorial Approach. *Combinatorica*, 22(3):409–416.

[Hujdurović et al., 2018] Hujdurović, A., Milanič, M., and Ries, B. (2018). Detecting Strong Cliques. *ArXiv e-prints*. `http://adsabs.harvard.edu/abs/2018arXiv180808817H`.

[Jeroslow et al., 1992] Jeroslow, R. G., Martin, K., Rardin, R. L., and Wang, J. (1992). Gainfree Leontief Substitution Flow Problems. *Mathematical Programming*, 57(1):375–414.

[Karbstein, 2013] Karbstein, M. (2013). *Line Planning and Connectivity.* Phd thesis, TU Berlin.

[Karp, 1972] Karp, R. M. (1972). Reducibility Among Combinatorial Problems. In *Complexity of Computer Computations*, pages 85–103. Springer.

[Kayll, 2010] Kayll, P. M. (2010). Kőnig–Egerváry Graphs are Non-Edmonds. *Graphs and Combinatorics*, 26(5):721–726.

[Keevash et al., 2015] Keevash, P., Knox, F., and Mycroft, R. (2015). Polynomial-Time Perfect Matchings in Dense Hypergraphs. *Advances in Mathematics*, 269:265–334.

[Keevash and Mycroft, 2015] Keevash, P. and Mycroft, R. (2015). *A Geometric Theory for Hypergraph Matching*, volume 233. American Mathematical Society.

[Kőnig, 1916] Kőnig, D. (1916). Über Graphen und ihre Anwendung auf Determinantentheorie und Mengenlehre. *Mathematische Annalen*, 77(4):453–465.

[Kőnig, 1934] Kőnig, D. (1932-1934). Über trennende Knotenpunkte in Graphen (nebst. Anwendungen auf Determinanten und Matrizen). *Acta Universitatis Szegediensis*, 6(2-3):155–179.

[Klamt et al., 2009] Klamt, S., Haus, U.-U., and Theis, F. (2009). Hypergraphs and Cellular Networks. *PLOS Computational Biology*, 5(5):1–6.

[Klimmek and Wagner, 1996] Klimmek, R. and Wagner, F. (1996). A Simple Hypergraph Min Cut Algorithm. Technical report, FU Berlin.

[Kovács, 2015] Kovács, P. (2015). Minimum-Cost Flow Algorithms: An Experimental Evaluation. *Optimization Methods and Software*, 30(1):94–127.

[Lakhal and Litzler, 1998] Lakhal, J. and Litzler, L. (1998). A Polynomial Algorithm for the Extendability Problem in Bipartite Graphs. *Information Processing Letters*, 65(1):11–16.

[Lawler, 1973] Lawler, E. L. (1973). Cutsets and Partitions of Hypergraphs. *Networks*, 3(3):275–285.

[Levit and Mandrescu, 2012] Levit, V. E. and Mandrescu, E. (2012). Critical Independent Sets and Kőnig–Egerváry Graphs. *Graphs and Combinatorics*, 28(2):243–250.

[Levit and Milanič, 2014] Levit, V. E. and Milanič, M. (2014). Equistable Simplicial, Very Well-Covered, and Line Graphs. *Discrete Applied Mathematics*, 165:205–212. 10th Cologne/Twente Workshop on Graphs and Combinatorial Optimization (CTW 2011).

[Liu and Liu, 2002] Liu, Y. and Liu, G. (2002). The Fractional Matching Numbers of Graphs. *Networks*, 40(4):228–231.

[Löbel, 1996] Löbel, A. (1996). Solving Large-Scale Real-World Minimum-Cost Flow Problems by a Network Simplex Method. Technical Report SC-96-07, ZIB, Takustr. 7, 14195 Berlin.

[Lovász, 1972] Lovász, L. (1972). Normal Hypergraphs and the Perfect Graph Conjecture. *Discrete Mathematics*, 2(3):253 – 267.

[Lovász, 1983] Lovász, L. (1983). Ear-Decompositions of Matching-Covered Graphs. *Combinatorica*, 3(1):105–117.

[Lovász, 1987] Lovász, L. (1987). Matching Structure and the Matching Lattice. *Journal of Combinatorial Theory, Series B*, 43(2):187–222.

[Lovász and Plummer, 1986] Lovász, L. and Plummer, M. D. (1986). *Matching Theory*, volume 29 of *Annals of Discrete Mathematics.* Elsevier.

[Lucchesi et al., 2018] Lucchesi, C., de Carvalho, M., Kothari, N., and Murty, U. (2018). On Two Unsolved Problems Concerning Matching Covered Graphs. *SIAM Journal on Discrete Mathematics*, 32(2):1478–1504.

[Mak and Wong, 2000] Mak, W.-K. and Wong, D. (2000). A Fast Hypergraph Min-Cut Algorithm for Circuit Partitioning. *Integration, the VLSI Journal*, 30(1):1–11.

[Marchand et al., 2002] Marchand, H., Martin, A., Weismantel, R., and Wolsey, L. (2002). Cutting Planes in Integer and Mixed Integer Programming. *Discrete Applied Mathematics*, 123(1):397–446.

[McAvaney et al., 1993] McAvaney, K., Robertson, J., and DeTemple, D. (1993). A Characterization and Hereditary Properties for Partition Graphs. *Discrete Mathematics*, 113(1):131–142.

[McCuaig, 2001] McCuaig, W. (2001). Brace Generation. *Journal of Graph Theory*, 38(3):124–169.

[Orlin, 1997] Orlin, J. B. (1997). A Polynomial Time Primal Network Simplex Algorithm for Minimum Cost Flows. *Mathematical Programming*, 78(2):109–129.

[Plummer, 1970] Plummer, M. D. (1970). Some Covering Concepts in Graphs. *Journal of Combinatorial Theory*, 8(1):91–98.

[Plummer, 1980] Plummer, M. D. (1980). On n-Extendable Graphs. *Discrete Mathematics*, 31(2):201–210.

[Plummer, 1993] Plummer, M. D. (1993). Well-Covered Graphs: A Survey. *Quaestiones Mathematicae*, 16(3):253–287.

[Plummer, 1994] Plummer, M. D. (1994). Extending Matchings in Graphs: A Survey. *Discrete Mathematics*, 127(1):277–292.

[Plummer, 2007] Plummer, M. D. (2007). Graph Factors and Factorization: 1985–2003: A Survey. *Discrete Mathematics*, 307(7):791–821. Cycles and Colourings 2003.

[Plummer, 2008] Plummer, M. D. (2008). Recent Progress in Matching Extension. In *Building Bridges: Between Mathematics and Computer Science*, pages 427–454. Springer Berlin Heidelberg, Berlin, Heidelberg.

[Qinglin, 1993] Qinglin, Y. (1993). Characterizations of Various Matching Extensions in Graphs. *Australasian Journal of Combinatorics*, 2:55–64.

[Reuther, 2017] Reuther, M. (2017). *Mathematical Optimization of Rolling Stock Rotations*. PhD thesis, Technische Universität Berlin. `https://depositonce.tu-berlin.de/handle/11303/6309`.

[Rödl et al., 2009] Rödl, V., Ruciński, A., and Szemerédi, E. (2009). Perfect Matchings in Large Uniform Hypergraphs with Large Minimum Collective Degree. *Journal of Combinatorial Theory, Series A*, 116(3):613–636.

[Sankaranarayana and Stewart, 1992] Sankaranarayana, R. S. and Stewart, L. K. (1992). Complexity Results for Well-Covered Graphs. *Networks*, 22(3):247–262.

[Scheidweiler and Triesch, 2016] Scheidweiler, R. and Triesch, E. (2016). A Note on the Duality between Matchings and Vertex Covers in Balanced Hypergraphs. *Journal of Combinatorial Optimization*, 32(2):639–644.

[Scheidweiler, 2011] Scheidweiler, R. B. (2011). *Matchings in Balanced Hypergraphs*. PhD thesis, RWTH Aachen. `http://publications.rwth-aachen.de/record/64351`.

[Schrijver, 2002] Schrijver, A. (2002). *Combinatorial Optimization: Polyhedra and Efficiency*, volume 24. Springer Science & Business Media.

[Seymour, 1980] Seymour, P. D. (1980). Decomposition of Regular Matroids. *Journal of Combinatorial Theory, Series B*, 28(3):305–359.

[Sterboul, 1979] Sterboul, F. (1979). A Characterization of the Graphs in which the Transversal Number Equals the Matching Number. *Journal of Combinatorial Theory, Series B*, 27(2):228–229.

[Sumner, 1979] Sumner, D. P. (1979). Randomly Matchable Graphs. *Journal of Graph Theory*, 3(2):183–186.

[Thakur and Tripathi, 2009] Thakur, M. and Tripathi, R. (2009). Linear Connectivity Problems in Directed Hypergraphs. *Theoretical Computer Science*, 410(27):2592–2618.

[Tutte, 1952] Tutte, W. (1952). The Factors of Graphs. *Canadian Journal of Mathematics*, 4(3):314–328.

[Tutte, 1954] Tutte, W. T. (1954). A Short Proof of the Factor Theorem for Finite Graphs. *Canadian Journal of Mathematics*, 6(1954):347–352.

[Valiant, 1979] Valiant, L. (1979). The Complexity of Computing the Permanent. *Theoretical Computer Science*, 8(2):189–201.

[Yannakakis, 1985] Yannakakis, M. (1985). On a Class of Totally Unimodular Matrices. *Mathematics of Operations Research*, 10(2):280–304.

[Zhang and Zhang, 2006] Zhang, F. and Zhang, H. (2006). Construction for Bicritical Graphs and k-Extendable Bipartite Graphs. *Discrete Mathematics*, 306(13):1415–1423.

[Zhao, 2016] Zhao, Y. (2016). Recent Advances on Dirac-Type Problems for Hypergraphs. In Beveridge, A., Griggs, J. R., Hogben, L., Musiker, G., and Tetali, P., editors, *Recent Trends in Combinatorics*, pages 145–165. Springer International Publishing.

Index