

Randomized Approximation for the Matching and Vertex Cover Problem in Hypergraphs

Complexity and Algorithms

Dissertation

zur Erlangung des akademischen Grades
Doktor der Naturwissenschaften
(Dr. rer. nat)

der Technischen Fakultät
der Christian-Albrechts-Universität zu Kiel

Dipl.-Math. Mourad El Ouali

Kiel

2013

1. Gutachter: Prof. Dr. Srivastav
Christian-Albrechts-Universität zu Kiel

2. Gutachter: Prof. Dr. Jäger
University of Umea,
Christian-Albrechts-Universität zu Kiel

Datum der mündlichen Prüfung: 30.08.2013

Zusammenfassung

Diese Arbeit befasst sich mit dem Entwurf und der mathematischen Analyse von randomisierten Approximationsalgorithmen für das Hitting Set Problem und das b -Matching Problem in Hypergraphen.

Zuerst präsentieren wir einen randomisierten Algorithmus für das Hitting Set Problem, der auf linearer Programmierung basiert. Mit diesem Verfahren und einer Analyse, die auf der probabilistischen Methode fußt, erreichen wir für verschiedene Klassen von Instanzen drei neue Approximationsgüten, die die bisher bekannten Ergebnisse (Krevilevich [1997], Halperin [2001]) für das Problem verbessern. Die Analysen beruhen auf Konzentrationsungleichungen für Summen von unabhängigen Zufallsvariablen aber auch Martingal-basierten Ungleichungen, wie die aus der Azuma-Ungleichung abgeleitete Bounded Difference-Inequality, in Kombination mit kombinatorischen Argumenten.

Für das b -Matching Problem in Hypergraphen analysieren wir zunächst seine Komplexität und erhalten zwei neue Ergebnisse.

Wir geben eine polynomielle Reduktion von einer Instanz eines geeigneten Problems zu einer Instanz des b -Matching-Problems an und zeigen ein Nicht-Approximierbarkeitsresultat für das Problem in uniformen Hypergraphen. Dieses Resultat verallgemeinert das Ergebnis von Safra et al. (2006) von $b = 1$ auf $b \in O\left(\frac{l}{\ln l}\right)$. Safra et al. zeigten, dass es für das 1-Matching Problem in uniformen Hypergraphen unter der Annahme $\mathcal{P} \neq \mathcal{NP}$ keinen polynomiellen Approximationsalgorithmus mit einer Ratio $O\left(\frac{l}{\ln l}\right)$ gibt.

Weiterhin beweisen wir, dass es in uniformen Hypergraphen mit beschränktem Knoten-Grad kein PTAS für das Problem gibt, solange $\mathcal{P} \neq \mathcal{NP}$.

Abstract

This thesis studies the design and mathematical analysis of randomized approximation algorithms for the hitting set and b -matching problems in hypergraphs.

We present a randomized algorithm for the hitting set problem based on linear programming. The analysis of the randomized algorithm rests upon the probabilistic method, more precisely on some concentration inequalities for the sum of independent random variables plus some martingale based inequalities, as the bounded difference inequality, which is a derived from Azuma inequality.

In combination with combinatorial arguments we achieve some new results for different instance classes that improve upon the known approximation results for the problem (Krevilevich (1997), Halperin (2001)).

We analyze the complexity of the b -matching problem in hypergraphs and obtain two new results.

We give a polynomial time reduction from an instance of a suitable problem to an instance of the b -matching problem and prove a non-approximability ratio for the problem in l -uniform hypergraphs. This generalizes the result of Safra et al. (2006) from $b = 1$ to $b \in O\left(\frac{l}{\ln l}\right)$. Safra et al. showed that the 1-matching problem in l -uniform hypergraphs can not be approximated in polynomial time within a ratio $O\left(\frac{l}{\ln l}\right)$, unless $\mathcal{P} = \mathcal{NP}$.

Moreover, we show that the b -matching problem on l -uniform hypergraphs with bounded vertex degree has no polynomial time approxi-

mation scheme (PTAS), unless $\mathcal{P} = \mathcal{NP}$.

Introduction

In this work we study the polynomial-time approximability of the b -matching and the hitting set problem in hypergraphs. Both problems are well explored in graph theory. For graphs, the hitting set problem is known as the vertex cover problem and it is \mathcal{NP} -hard [48], whereas the b -matching problem is solvable in polynomial time [62]. On the other hand, the b -matching problem in hypergraphs is a generalization of the well-known set packing problem or simply 1-matching problem in hypergraphs and it is \mathcal{NP} -hard, too [48]. The terminology „vertex cover“ and „hitting set“ is used for hypergraphs in a synonymous way.

The Hitting Set Problem

Let $\mathcal{H} = (V, \mathcal{E})$ be a hypergraph. A hitting set (or vertex cover) in \mathcal{H} is a set $C \subseteq V$ in which all edges are incident. The hitting set problem is to find a hitting set of minimum cardinality. The problem is \mathcal{NP} -hard, even for graphs [48]. Thus the research has focused on polynomial-time approximation algorithms.

For graphs it is known that a 2-factor approximation can be achieved in polynomial time, for example by the primal-dual method [41], but an approximation better than factor of 2 is not possible, if we assume the unique game conjecture [52]. For graphs with maximum degree bounded by a constant Δ several ρ -improvements with $\rho < 2$ were proved (Ghandi, Khuler, Srinivasan (2000), Halperin (2001)). For

Introduction

hypergraphs with hyperedge size at most l , a greedy algorithm leads to a factor l approximation, while an approximation better than l is not possible, if we assume the unique game conjecture [52].

As hypergraphs are a generalization of graphs, a key question arises as what extent the achieved approximation results for the vertex cover problem in graphs can be generalized to hypergraphs.

Thus, it is an important problem to characterize classes of hypergraphs for which the approximation barrier of l may be broken. Krivelevich [55] proved an approximation ratio of $l(1 - cn^{\frac{1-l}{l}})$ for l -uniform hypergraphs. Halperin [36] showed that the problem can be approximated within a factor of $l - (1 - o(1))\frac{l(l-1)\ln\ln n}{\ln n}$ for l -uniform hypergraphs with $l^3 = o(\frac{\ln\ln n}{\ln\ln\ln n})$. Note that this condition enforces $n = 2^{2^l}$ and thus applies only to astronomically large sized hypergraphs.

We present a randomised algorithm of *hybrid type* that combines LP-based randomised rounding, sparsening of the hypergraph and greedy repairing. The randomized rounding technique computes an initial integer solution which might be infeasible. We use a greedy approach to make the solution feasible. The mathematical challenge is to analyze the combination of randomized rounding and the greedy repairing. With an elaborate analysis based on the probabilistic method adapted to different instance classes, mainly characterized by parameters like the maximal edge size l and the maximum vertex degree Δ , we achieve three new results that improve the previous approximations.

For hypergraphs with maximum hyperedge size l and maximum vertex degree Δ (not necessarily assumed to be constants) we show that our algorithm achieves for $l = O(\sqrt{n})$ and $\Delta = O(n^{\frac{1}{4}})$ an approximation ratio of $l(1 - \frac{c}{\Delta})$ with constant probability, for some constant $c > 0$,

improving the bounds obtained by Krivelevich [55].

For the class of uniform, quasi-regularisable hypergraphs, which are known and useful in the combinatorics of hypergraphs (see definition in Chapter 2, for more details see Berge [10]) we prove an approximation ratio of $l(1 - \frac{n}{8m})$ under the assumption that $\Delta = O(n^{\frac{1}{3}})$.

Moreover, we consider hypergraphs, where l and Δ are constants, and achieve a ratio of $l(1 - \frac{l-1}{4\Delta})$, which is an improvement of the bound of $l(1 - c\Delta^{\frac{1}{1-l}})$ presented by Krivelevich [55] and the bound of $l - (1 - o(1))\frac{l(l-1)\ln\ln\Delta}{\ln\Delta}$ for $l^3 = o(\frac{\ln\ln\Delta}{\ln\ln\ln\Delta})$ given by Halperin [36].

Table 1. Summary of the results for the hitting set problem

Hypergraph	Previous Results	Our Results
$l = O(\sqrt{n})$	Krivelevich: $l(1 - cn^{\frac{1-l}{l}})$	$l(1 - c/n^{\frac{1}{4}})$
quasi-regularisable	—	$l(1 - \frac{n}{8m})$
l and Δ constants	Krivelevich: $l(1 - c\Delta^{\frac{1}{1-l}})$ Halperin: $l - (1 - o(1))\frac{l(l-1)\ln\ln\Delta}{\ln\Delta}$	$l(1 - \frac{l-1}{4\Delta})$

Furthermore, we present a hybrid randomized algorithm for the partial vertex cover problem for hypergraphs with constant maximum hyper-edge degree D and maximum hyperedge size l (not necessarily assumed to be constant). A similar analysis as for the hitting set problem yields a partial vertex cover of cardinality at most $l(1 - \Omega(1/(D+1)))\text{Opt}$. To our best knowledge this is the first approximation ratio below the approximation barrier of l .

Table 2. Summary of the results for the partial vertex cover Problem

Hypergraph	Previous Results	Our Results
D constant	Srinivasan et al. : l	$l(1 - \Omega(1/(D + 1)))$
$k \geq \frac{m}{4}$ and Δ constant	—	$l\left(1 - \Omega\left(\frac{1}{\Delta}\right)\right)$

The b -Matching Problem in Hypergraphs

Let $\mathcal{H} = (V, \mathcal{E})$ be a hypergraph. For a given $b \in \mathbb{N}$ we call a set $M \subseteq \mathcal{E}$ a b -matching if no vertex in V is contained in more than b edges of M . Maximum b -matching is the problem of finding a b -matching with maximum cardinality. For graphs the problem is solvable in polynomial-time [62]. But already for 3-uniform hypergraphs it becomes \mathcal{NP} -hard, and there has been a large body of work concerning polynomial-time approximation algorithms [57, 68, 74, 76]. We consider the question as to how the parameter b influences the approximability of the problem. We give a polynomial-time reduction from an instance of a suitable \mathcal{NP} -hard problem to an instance of the b -matching problem and prove that it is \mathcal{NP} -hard to approximate the b -matching problem in an l -uniform hypergraph within any ratio smaller than $O\left(\frac{l}{b \ln l}\right)$.

This shows that the approximation depends on b and the non-approximability ratio tends towards a constant, if b tends to $\frac{l}{\ln l}$. In other words, if b is getting large, a much better approximation ratio might be possible. This result is a generalization of the result of Safra et al. [37], who showed that the 1-matching problem in l -uniform hypergraphs can not be approximated in polynomial time within a ratio of $O\left(\frac{l}{\log l}\right)$, unless $\mathcal{P} = \mathcal{NP}$. It is notable that the proof

of Safra et al. in [37] cannot be lifted to $b \geq 2$. In fact some new techniques and ingredients are required: e.g. the probabilistic proof of the existence of a hypergraph with 'almost' disjoint b -matchings, where dependent events have to be decoupled (in contrast to Safra et al.) and the generation of some sparse hypergraph.

Furthermore, we show that the b -matching problem in l -uniform hypergraphs admits no polynomial-time approximation scheme, unless $\mathcal{P} = \mathcal{NP}$. This generalizes a result of Kann [47] for the 1-matching problem in uniform hypergraphs. We also give a reduction that preserves approximability between this problem and the set multicover problem in Δ -regular hypergraphs. This means, if we can prove that one of the problems has no PTAS, then the same holds for the other problem.

Outline of the Thesis

The thesis is organized as follows:

In Chapter 1, we state some definitions and notations, describe some techniques and methods that will be applied in further chapters in this thesis.

In Chapter 2, we present a LP-based algorithm for the hitting set problem in hypergraphs that we analyze in two different ways for different classes of hypergraphs and derive three new results.

In Chapter 3, we study the inapproximability aspect of the b -matching problem in uniform hypergraphs. First we introduce notations, give definitions and review some known results. Then, we give

Introduction

an \mathcal{NP} -reduction to a b -matching instance proving an inapproximability result for this problem.

Chapter 4, treats a further aspect of the hardness of the b -matching problem in uniform hypergraphs. We prove that the problem has no polynomial-time approximation, unless $\mathcal{P} = \mathcal{NP}$. To this end, we first give the required notions and definitions and then present a linear reduction that constructs a suitable instance of the problem and prove its correctness.

Contents

Introduction	v
Acknowledgements	xiii
List of Abbreviations	xv
1 Preliminaries about Definitions and Concepts	1
1.1 Hypergraphs and Problems	1
1.2 The Notion of Approximation and Randomized Algorithms	5
1.2.1 Linear Programming and Randomized Rounding	9
1.3 Probabilistic Tools	12
2 The Hitting Set Problem	15
2.1 The Hitting Set Problem and Previous Work	16
2.2 The Randomized Algorithm	18
2.3 Two-Step Analysis of the Algorithm	20
2.4 One-Step Analysis of the Algorithm	31
2.5 Partial Vertex Cover in Hypergraphs	39
2.6 Summary and Further Work	43
3 Inapproximability of b-Matching in l-Uniform Hypergraphs	47
3.1 Problem and Related Work.	48
3.2 Definitions and complexity-theoretic Tools	50
3.3 The Construction and the Inapproximability Result.	53

Contents

3.4	Proof of the Structure Theorem	64
3.5	Summary and Future Work	79
4	The b-Matching Problem in l-Uniform Hypergraphs is MAX-SNP-Hard	83
4.1	Tools and Definitions	84
4.2	Main Result	87
4.3	Analysis of the Reduction	93
4.4	The Extension to l -uniform hypergraphs	100
4.5	Further L -reduction	101
4.6	Summary and Further Work.	104

Acknowledgements

I would like to thank all the people who have been with me during all these years of hard works for their support and confidence in me.

I thank my doctoral advisor Prof. Dr. Anand Srivastav for providing me with the research opportunities and for his guidance, inspiration and gentle throughout my studies.

I thank Prof. Dr. Gerold Jäger for discussion on hardness and approximability of the b-matching problem in hypergraphs.

I thank Antje Fretwurst for joint research on the inapproximability of the b-matching problem in uniform hypergraphs.

I thank Helena Fohlin for joint research on the vertex cover problem in hypergraphs.

I thank Kati & Felix Land, Dr. Barbara Langfeld, Peter Mustermann, Dr. Volkmar Sauerland and Mayank Singhal for their help by thoroughly checking the proofs.

List of Abbreviations

$[n]$	the set $\{1, \dots, n\} \subset \mathbb{N}$
\mathbb{Z}	the set of integers
\mathbb{N}	the set of natural numbers
$\mathbb{Q}_{\geq 0}$	the set of positive rational numbers
$\mathbb{R}_{\geq 0}$	the set of positive real numbers
(a, b)	the set of real numbers strictly greater than a and strictly smaller than b
$\Pr(X)$	the probability of a random variable X
$\mathbb{E}(X)$	the expectation of a random variable X
$\text{Var}(X)$	the variance of a random variable X
\ln	\log_e , i.e. the natural logarithm
$\nu_{\mathbf{b}}(\mathcal{H})$	the size of a maximum \mathbf{b} -matching set of \mathcal{H} , for $\mathbf{b} \in \mathbb{N}^n$
$\rho_{\mathbf{k}}(\mathcal{H})$	the size of a minimum set \mathbf{k} -multi cover of \mathcal{H} , for $\mathbf{k} \in \mathbb{N}^n$

List of Figures

1.1	Rank and degree of hypergraphs	4
3.1	First phase of the reduction	57
3.2	Second phase of the reduction	57
4.1	An example of ring triples	88
4.2	An example of binary trees	88
4.3	An example of binary trees, its adjacent clause triple and ring triples	89
4.4	An example of connecting the elements between two symmetric trees	90
4.5	An example of element labelling in a tree with two levels	91
4.6	An example of the final instance	92
4.7	An example of an optimal standard matching	95

List of Tables

1	Summary of the results for the hitting set problem . .	vii
2	Summary of the results for the partial vertex cover Problem	viii

Preliminaries about Definitions and Concepts

Matching and covering problems in graphs and hypergraphs are fundamental tasks in combinatorial optimization. In this thesis we focus on hypergraphs. In this chapter we will introduce some elementary definitions and notions that we will frequently be used in the upcoming chapters. More technical definitions and preliminaries will be introduced in those chapters where they are required for results and analysis, respectively. The definitions in this chapter can also be found in the books of Korte [54], Wegener [32], Berge [10] and Hromkovič [42].

1.1 Hypergraphs and Problems

A hypergraph \mathcal{H} is an ordered pair (V, \mathcal{E}) where $V = \{v_1, \dots, v_n\}$ is a finite set and $\mathcal{E} = \{E_1, \dots, E_m\}$ is a collection of non-empty subsets of V . The elements v_1, \dots, v_n of V are called vertices and the sets E_1, \dots, E_m are called hyperedges. The size or the cardinality $|E|$ of a hyperedge $E \in \mathcal{E}$ is the number of vertices in E . Hypergraphs are a generalization of simple graphs, since a simple graph is a hypergraph with all hyperedges having cardinality 2. A hypergraph can be

1. Preliminaries about Definitions and Concepts

represented by its incidence matrix

$A = (a_{ij}) \in \{0, 1\}^{n \times m}$, where $a_{ij} = 1$ if $v_i \in E_j$, and 0 otherwise, so the columns of the incidence matrix represent the hyperedges E_1, \dots, E_m and the rows represent the vertices v_1, \dots, v_n .

We say that a vertex u is a neighbor of a vertex v , if there exist a hyperedge $E \in \mathcal{E}$ that contains both u and v . The set of neighbors of a vertex v is denoted by $N(v)$. More generally, for $U \subseteq V$, we denote by $N(U)$ the set of all neighbors in $V \setminus U$ of vertices in U . The formal definition is:

$$\begin{aligned} N(v) &= \{u \in V \setminus \{v\} : \exists E \in \mathcal{E}, \{u, v\} \subseteq E\} \\ N(U) &= \{v \in V \setminus U : \exists u \in U, v \in N(u)\}. \end{aligned}$$

Further, for a set $U \subseteq V$ we denote by

$$\Gamma(U) := \{F \in \mathcal{E} ; U \cap F \neq \emptyset\}$$

the set of hyperedges incident to the set U .

A hypergraph \mathcal{H} has vertex- (resp. hyperedge) weights if to every vertex (resp. hyperedge) a weight is assigned. Formally, a vertex-weight (resp. hyperedge weight) of a hypergraph $\mathcal{H} = (V, \mathcal{E})$ is a function $w : V \rightarrow \mathbb{R}$ (resp. $w : \mathcal{E} \rightarrow \mathbb{R}$). Let us define for $U \subseteq V$, $w(U) := \sum_{v \in U} w(v)$ resp. for $\mathcal{F} \subseteq \mathcal{E}$ $w(\mathcal{F}) := \sum_{E \in \mathcal{F}} w(E)$.

In this thesis we are essentially interested in the class of simple hypergraphs, also called Sperner family [10], where all hyperedges are distinct. We will study instances that are characterized by the following three parameters:

The rank of a hypergraph. The rank $r(\mathcal{H})$ of a hypergraph $\mathcal{H} = (V, \mathcal{E})$ is the maximum hyperedge size, namely $\max\{|E| : E \in \mathcal{E}\}$. A hyper-

1.1. Hypergraphs and Problems

graph is called l -uniform if all hyperedges have the same cardinality l . Thus a graph is a 2-uniform hypergraph. A further subclass are hypergraphs with bounded rank where the size of every hyperedge is at most a given constant l .

The vertex degree. For $v \in V$ define the (vertex-) degree $\deg(v)$ of v as the cardinality of $\Gamma(\{v\})$ so $\deg(v) = |\Gamma(\{v\})|$. We say, \mathcal{H} is r -regular, if $\deg(v) = r$ for all $v \in V$.

The hyperedge degree. For hyperedge $E \in \mathcal{E}$ we denote by $\deg(E)$ the hyperedge degree of E , defined by

$$\deg(E) := |\Gamma(E)| = |\{F \in \mathcal{E} \setminus E; E \cap F \neq \emptyset\}|.$$

In the rest of the thesis we will denote by

$$\Delta(\mathcal{H}) := \max_{v \in V} \deg(v)$$

the maximum vertex degree, by

$$\bar{d}(\mathcal{H}) := \frac{\sum_{v \in V} \deg(v)}{|V|}$$

the average vertex degree and by

$$D(\mathcal{H}) := \max_{E \in \mathcal{E}} \deg(E)$$

the maximum hyperedge degree.

The notion of the degree of a hypergraph allows us to define some important hypergraph classes: hypergraphs with bounded vertex degree (for a given constant $\Delta \in \mathbb{N}$, the degree of every vertex is at most Δ), hypergraphs with bounded hyperedge degree (for a given

1. Preliminaries about Definitions and Concepts

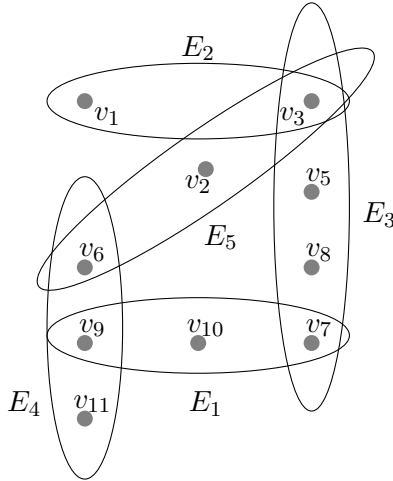


Figure 1.1. An example of a hypergraph with rank 4: $|E_3| = 4$, maximum degree 3: $\deg(v_3) = 3$ and maximum hyperedge-degree 3: $\deg(E_3) = 3$.

constant $D \in \mathbb{N}$, the degree of every hyperedge is at most D), regular hypergraphs (all vertices have the same degree Δ).

We investigate the following three fundamental problems in combinatorial optimization:

b- Matching Problem in Hypergraphs: For a given hypergraph $\mathcal{H} = (V, \mathcal{E})$, $\mathbf{b} = (b_1, b_2, \dots, b_n) \in \mathbb{N}^n$ and $w : \mathcal{E} \rightarrow \mathbb{Q}_{\geq 0}$, we call a set $M \subseteq \mathcal{E}$ a **b-matching** in \mathcal{H} , if no vertex $i \in V$ is contained in more than b_i hyperedges of M .

The **b-matching** problem in hypergraphs is the task of finding a maximum weight **b-matching** in \mathcal{H} .

An important special case of **b-matching** in hypergraphs is the l -dimensional **b-matching** defined as follows:

l -Dimensional b-Matching Problem: This problem is a variant of **b-**

1.2. The Notion of Approximation and Randomized Algorithms

matching in l -uniform hypergraphs, where the vertices of the input hypergraph are a union of l disjoint sets, $V = V_1 \dot{\cup} V_2 \dots \dot{\cup} V_l$, and each hyperedge E contains exactly one vertex from each set such that $E \subseteq V_1 \times V_2 \times \dots \times V_l$ and $|E| = l$.

The l -dimensional \mathbf{b} -matching problem is the task of finding a maximum weight l -dimensional \mathbf{b} -matching in \mathcal{H} .

We will mainly be concerned with the case that $b_i = b$ for some scalar value $b \geq 1$ for all $i \in V$.

Hitting Set Problem: For a given hypergraph $\mathcal{H} = (V, \mathcal{E})$ and $w : V \rightarrow \mathbb{Q}_{\geq 0}$, we call a set $C \subseteq V$ a hitting set in \mathcal{H} if for all hyperedges it holds that they are incident in some vertex of C . The hitting set problem is the task of finding a hitting set of minimum weight.

Set Multicover Problem: For a given hypergraph $\mathcal{H} = (V, \mathcal{E})$ and $\mathbf{k} = (k_1, k_2, \dots, k_n) \in \mathbb{N}^n$ and $w : V \rightarrow \mathbb{Q}_{\geq 0}$, we call a subset $\text{SMC} \subseteq \mathcal{E}$ a set multicover in \mathcal{H} if no vertex $i \in V$ is contained in less than k_i hyperedges of SMC . The set multicover problem is the task of finding a set multicover of minimum weight.

1.2 The Notion of Approximation and Randomized Algorithms

We briefly recall the concept of approximation algorithms.

Let Π be an optimization problem with feasible solutions in $\mathbb{Q}_{\geq 0}$. For an input instance I of Π we denote by $\text{Opt}(I)$ the value of an optimal solution. We distinguish between two kinds of approximation algorithms. The first kind returns solutions that differ from the optimal

1. Preliminaries about Definitions and Concepts

solution by a constant and is called an absolute approximation algorithm or approximation algorithm with additive constant. Formally, we have the following

1.1 Definition. An absolute approximation algorithm for an optimization problem Π is a polynomial-time algorithm A for which there exists a constant $r > 0$ such that

$$|A(I) - \text{Opt}(I)| \leq r$$

for all instances I of Π , where $A(I)$ is the objective value of the solution given by A . In this case we call r an additive approximation error.

The second kind is called an algorithm with an approximation ratio or multiplicative ratio and is defined as follows

1.2 Definition. Let $r \geq 1$. An r -approximation algorithm for a maximization problem Π is a polynomial-time algorithm A such that

$$A(I) \geq \frac{1}{r} \text{Opt}(I),$$

for all instances I of Π . $\frac{1}{r}$ is called approximation factor and r is called the approximation ratio.

Analogously, we define an r -approximation algorithm for a minimization problem Π to be a polynomial-time algorithm A such that

$$A(I) \leq r \text{Opt}(I).$$

r is called approximation factor and $\frac{1}{r}$ is the approximation ratio.

Approximation Schemes. As mentioned above, problems that are \mathcal{NP} -

1.2. The Notion of Approximation and Randomized Algorithms

complete cannot be solved in polynomial-time, unless $\mathcal{P} = \mathcal{NP}$, but some of them may admit a polynomial-time approximation algorithm with a ratio of $1 + \epsilon$ for each $\epsilon > 0$. We call such an algorithm a polynomial-time approximation scheme.

1.3 Definition. Let Π be an optimization problem with non-negative weights. A polynomial-time approximation scheme (PTAS) for Π is an algorithm A that takes as input an instance I of Π and an $\epsilon > 0$ such that, for each *fixed* ϵ , A is a $(1 + \epsilon)$ -approximation algorithm for Π . The running time of A can be bounded by a function that is polynomial in the instance size $|I|$.

Note that the running time of a PTAS may depend exponentially in ϵ , e.g. $O(|I|^{\frac{1}{\epsilon}})$.

Hardness of Optimization Problems. For many problems it is possible to prove that even the existence of an r -approximation algorithm with small r is impossible, unless $\mathcal{P} = \mathcal{NP}$. Such results are called non-approximability results. The objective of this section is to introduce some techniques that, under assumptions like $\mathcal{P} \neq \mathcal{NP}$, prove the non-approximability of optimization problems. We provide different results for the different problems under consideration. On the one hand, we ask for the non-existence of a PTAS. On the other hand, we prove lower bounds for the approximation ratio of any polynomial-time approximation algorithm. To obtain such hardness results there are in general three methods:

Reduction to \mathcal{NP} -hard decision problems. Let Π be the problem for which we want to find a feasible solution within a fixed approximation ratio r .

The aim of the technique is to reduce an \mathcal{NP} -hard problem to the

1. Preliminaries about Definitions and Concepts

problem Π and show that Π does not admit any polynomial-time algorithm with approximation ratio r , under the assumption $\mathcal{P} \neq \mathcal{NP}$.

Approximation-preserving reduction. In general the approximation properties of a \mathcal{NP} -hard problem will not be preserved under a \mathcal{NP} -reduction.

The concepts of \mathcal{NP} -completeness and that of approximation-preserving reduction are similar. While the first concept, under the assumption $\mathcal{P} \neq \mathcal{NP}$, provides a criterion to prove that a given problem does not belong to the class \mathcal{P} , the second concept provides under the same assumption an argument to show that an optimization problem does not admit a PTAS. Most of the approximation-preserving reductions are based on the following idea: Let f be a approximation-preserving reduction from an optimization problem Π_1 to another optimization problem Π_2 . Then, the two following criteria must be satisfied: first, f must transform each instance of Π_1 to an instance of Π_2 , secondly, it must be possible to transform every solution of the constructed instance of Π_2 to a solution for the original instance.

In this thesis we will be concerned only with the so-called "linear reduction". A formal definition of a linear reduction is given in Chapter 4.

Application of the PCP-Theorem. Several inapproximability results are based on one of the deepest and hardest results in the theory of computation, the so called PCP¹-Theorem [5, 6], which gave a new characterization of the class \mathcal{NP} .

In this thesis, we are essentially interested in the first two methods, which we will explicitly describe and apply in Chapters 3 and 4.

Randomized Algorithms. A randomized algorithm is informally speaking

¹the letters PCP stand for "probabilistically checkable proofs"

1.2. The Notion of Approximation and Randomized Algorithms

an algorithm which uses random bits, e.g. cast a coin or a dice etc. during its execution. Algorithmic problems can often be solved very simply and efficiently by a randomized algorithm. Thus, in contrast to deterministic algorithms, the behaviour of a randomized algorithm depends not only on its input, but also on random bits.

Consequently if we run a randomized algorithm A on the same input several times, we will very likely get different results. The difference may concern both the solution and the running time. Probability theory is used to prove a typical performance of the algorithms with (desirably) high probability.

1.2.1 Linear Programming and Randomized Rounding

Linear Programming. In this thesis we will frequently use linear programming relaxations of integer linear programs. A linear programming problem can be defined as the problem of optimizing (maximizing or minimizing) a linear function over \mathbb{Q}^n subject to a set of feasible solutions that is described by finitely many linear inequality and equality constraints.

Formally, let $A \in \mathbb{Q}^{m \times n}$ be a matrix and $b \in \mathbb{Q}^m$, $c \in \mathbb{Q}^n$ be two column vectors. A linear programming problem is the problem of finding a vector $x \in \mathbb{Q}^n$ that maximizes (or minimizes) the function $c^T x$ and satisfies $Ax \leq b$ (resp. $Ax \geq b$).

A linear program (LP) is an instance of a linear programming problem. If we replace the set \mathbb{Q}^n by \mathbb{Z}^n , we call the corresponding problem an integer linear programming problem. An instance of this problem type is called integer linear program (ILP). Formally, an ILP (maximization

1. Preliminaries about Definitions and Concepts

version) is given by

$$\max\{c^T x : x \in \mathbb{Z}^n, Ax \leq b\}.$$

By taking \mathbb{Q}^n instead of \mathbb{Z}^n , we relax the ILP and call this linear program the LP-relaxation. A relaxed optimal solution can be found in polynomial-time [46, 50], while solving ILPs is a \mathcal{NP} -hard problem [48]. An important case of an integer linear program is the 0/1-ILP where the variables are binary, i. e. from $\{0, 1\}$.

Let us consider a maximization problem. One of the fundamental results in the theory of mathematical programming is the duality theorem due to von Neumann [78] and Gale, Kuhn and Tucker [29].

1.4 Theorem (The Duality Theorem). *Let $P = \{x \in \mathbb{Q}_{\geq 0}^n : Ax \leq b\}$ and $D = \{y \in \mathbb{Q}_{\geq 0}^m : A^T y \geq c\}$ be nonempty sets. P is the primal and D is its dual program. It holds that*

$$\max\{c^T x : x \in P\} = \min\{b^T y : y \in D\}$$

The direction $\max\{c^T x : x \in P\} \leq \min\{b^T y : y \in D\}$ of the equality is easy to prove and is called the weak duality.

A large family of combinatorial optimisation problems, among them problems on graphs and hypergraphs, can be expressed as covering or packing integer programs.

Covering Problem. Covering problems are minimisation problems which can be expressed by an integer linear program (CIP) as follows:

$$\begin{array}{ll} \min & c^T x \\ \text{(CIP)} & Ax \geq b \\ & x \in \mathbb{Z}_{\geq 0}^n \end{array}$$

1.2. The Notion of Approximation and Randomized Algorithms

where A , b and c are as given above.

Among the problems that are treated in this thesis, the hitting set problem, the partial vertex cover problem in hypergraphs can be expressed as covering problems.

Packing Problems. Packing problems are maximization problems with an integer linear program (PIP) formulation as follows: Let A , b and c as above, then

$$\begin{aligned} \text{(PIP)} \quad & \max c^T x \\ & Ax \leq b \\ & x \in \mathbb{Z}_{\geq 0}^n \end{aligned}$$

The b -matching problem in hypergraphs treated in this thesis is a packing problem.

Randomized Rounding. A technique introduced by Raghavan and Thompson [68] that combines linear programming and randomized algorithms is referred to as *randomized rounding*. It works roughly as follows: After solving the LP relaxation of a binary ILP we aim to round the optimal fractional variables to 0 or 1 in order to get a feasible solution of the ILP which should be close to the integer optimum. For an optimal fractional solution $x = (x_1, \dots, x_n)$ of an LP, we generate a random variable $X = (X_1, \dots, X_n)$ by setting $X_i = 1$ with probability x_i and $X_i = 0$ with probability $1 - x_i$ for each i , independently for all i . The idea of this technique is to consider the optimal fractional values of a LP- solution as probabilities for the rounding process. Clearly, if a fractional value is 1 (or 0) then it will be picked (not picked) for the corresponding integer solution. Depending on the optimization problem under consideration, the

1. Preliminaries about Definitions and Concepts

rounding process can be varied in a skilful manner (see Chapter 2).

1.3 Probabilistic Tools

For the one-sided deviation the Chebychev-Cantelli inequality will be frequently used:

1.5 Theorem (see [1]). *Let X be a non-negative random variable with finite mean $\mathbb{E}(X)$ and variance $\text{Var}(X)$. Then for any $a > 0$ we have*

$$\Pr(X \geq \mathbb{E}(X) + a) \leq \frac{\text{Var}(X)}{\text{Var}(X) + a^2}.$$

A further useful concentration result is the independent bounded differences inequality theorem:

1.6 Theorem (see [63]). *Let $X = (X_1, X_2, \dots, X_n)$ be a family of independent random variables with X_k taking values in a set A_k for each k . Suppose that the real-valued function f defined on $A_1 \times A_2 \times \dots \times A_n$ satisfies $|f(x) - f(x')| \leq c_k$ whenever the vector x and x' differ only in the k -th coordinate. Let $\mathbb{E}(f(X))$ be the expected value of the random variable $f(X)$. Then for any $t > 0$ it holds*

$$\Pr(f(X) \leq \mathbb{E}(f(X)) - t) \leq \exp\left(\frac{-2t^2}{\sum_{k=1}^n c_k^2}\right).$$

The following estimate on the variance of a sum of dependent random variables can be proved as in the book of Alon and Spencer [3].

1.7 Lemma (see [3]). *Let X be the sum of finitely many 0/1 random variables, i. e., $X = X_1 + \dots + X_n$, and let $p_i = \mathbb{E}(X_i)$ for all $i = 1, \dots, n$. For a pair $i, j \in \{1, \dots, n\}$ we write $i \sim j$, if X_i and*

1.3. Probabilistic Tools

X_j are dependent. Let Γ be the set of all unordered dependent pairs i, j , i. e., 2-element sets $\{i, j\}$, and let $\gamma = \sum_{\{i,j\} \in \Gamma} \mathbb{E}(X_i X_j)$, then it holds

$$\text{Var}(X) \leq \mathbb{E}(X) + 2\gamma.$$

For a sum of independent random variables we will use the large deviation inequality due to Anghin and Valiant [63]:

1.8 Theorem (see [63]). *Let X_1, \dots, X_n be independent 0/1-random variables and $\mathbb{E}(X_i) = p_i$ for all $i = 1, \dots, n$. Let $X = \sum_{i=1}^n X_i$ and $\mu = \mathbb{E}(X)$. For any $\beta > 0$ it holds*

- (i) $\Pr(X \geq (1 + \beta) \cdot \mu) \leq \exp\left(-\frac{\beta^2 \mu}{2(1+\beta/3)}\right)$ and
- (ii) $\Pr(X \leq (1 - \beta) \cdot \mu) \leq \exp\left(-\frac{\beta^2 \mu}{2}\right)$.

The Hitting Set Problem

In this chapter¹ we analyze the hitting set problem, which is a generalization of the vertex cover problem in graphs.

The hitting set problem is one of Karp's 21 \mathcal{NP} -complete problems [48]. It is known that an approximation ratio of l , where l is the maximum edge size, can be easily achieved using a maximal matching. On the other hand, for constant l , an approximation ratio better than l cannot be achieved in polynomial time under the unique games conjecture [52]. Thus, breaking the l -barrier for significant classes of hypergraphs is a complexity-theoretically and algorithmically interesting problem, which has been studied by several authors. Here we will study the approximability aspect of the problem.

The chapter is organized as follows: In Section 2.1 we present the problem and state some results from earlier works. In Section 2.2 we propose a randomized hybrid algorithm for the hitting set problem, which combines LP-based randomized rounding, graph sparsening and greedy repairing. We analyze the algorithm for different instance classes. In Section 2.3 we analyze the approximation ratio for hypergraphs with non-constant hyperedge size and non-constant vertex degree. In Section 2.4 we analyze the algorithm in a different way and prove an approximation ratio for the subclass of uniform quasi-

¹This chapter is mainly based on the papers [20, 21]

2. The Hitting Set Problem

regularisable hypergraphs (Section 2.4.1) and uniform hypergraphs with bounded vertex degree (Section 2.4.2). In Section 2.5 we give a brief overview of the generalization of the hitting set problem and cite some important results concerning this problem. In Section 2.6 we give a summary of the chapter and discuss future work.

2.1 The Hitting Set Problem and Previous Work

Let $\mathcal{H} = (V, \mathcal{E})$ be a hypergraph with $n := |V|$ and $m := |\mathcal{E}|$.

Approximability of the Problem. The vertex cover problem and the well-known set cover problem are equivalent. This can be checked by changing the roles of vertices and hyperedges. Both problems have been explored extensively in the context of polynomial-time approximations.

Concerning the hardness aspect, it is known that under the unique games conjecture (UGC) the problem cannot be approximated within any *constant* factor better than 2 [52]. For general graphs the best known algorithms are due to Monien and Speckenmeyer [64] and Bar-Yehuda and Even [8]. They independently gave an algorithm that achieves a ratio of $(2 - \frac{\ln \ln n}{2 \ln n})$. The bounded variant of the vertex cover problem, i. e., where the graph has a bounded vertex degree Δ , is known to admit approximation algorithms better than their general versions, the quality of the approximation being a function of Δ . Hochbaum [41] gave a $(2 - \frac{2}{\Delta})$ -approximation and later Halldorsson and Radhakrishnan [34] obtained an improved $(2 - \frac{\ln \Delta + O(1)}{\Delta})$ -approximation. Using semidefinite programming, Halperin [36] showed that this problem can be approximated within a factor of $(2 - (1 - o(1)) \frac{2 \ln \ln \Delta}{\ln \Delta})$.

2.1. The Hitting Set Problem and Previous Work

The earliest published approximation algorithms for the hitting set problem achieve an approximation ratio of the order $\ln m + 1$ [14, 45, 57] by using a greedy heuristic, which gives a $\ln n + 1$ approximation for the set cover problem. A number of inapproximability results are known for the hitting set problem in general hypergraphs. Lund and Yannakakis [58] proved for the set cover problem that for any $\alpha < \frac{1}{4}$, the existence of a polynomial-time $(\alpha \ln n)$ -ratio approximation algorithm would imply that \mathcal{NP} has a quasipolynomial, i. e., $n^{\mathcal{O}(\text{poly}(\ln n))}$ time deterministic algorithm. This result was improved to $(1 - \varepsilon) \ln n$ for $\varepsilon \in (0, 1)$ by Feige [23]. A $c \cdot \ln n$ -approximation under the assumption that $\mathcal{P} \neq \mathcal{NP}$ was established by Safra and Raz [70], where c is a constant. A similar result for larger values of c was proved by Alon, Moshkovitz and Safra [2].

The hitting set problem remains hard for many hypergraph classes. Most interesting are l -uniform hypergraphs with a constant l , because, for them, under the unique game conjecture (UGC), it is \mathcal{NP} -hard to approximate the problem within a factor of $l - \varepsilon$, for any *fixed* $\varepsilon > 0$, see [52], while an approximation ratio of l can be easily achieved by finding a maximal matching. Therefore, the problem of breaking the l -barrier for significant and interesting classes of hypergraphs received much attention.

For l -uniform hypergraphs, several authors achieved the ratio of l using different techniques (see, e. g., [7, 30, 38, 41]). The first and important result breaking the barrier of l for l -uniform hypergraphs is due to Krivelevich [55]. He proved an approximation ratio of $l(1 - cn^{\frac{1-l}{l}})$, for some constant $c > 0$, using a combination of the LP-based algorithm and the local ratio approach described by Bar-Yehuda and Even [8]. Later, for l -uniform hypergraphs with

2. The Hitting Set Problem

$l^3 = o(\frac{\ln \ln n}{\ln \ln \ln n})$ and $\Delta = O(n^{l-1})$, Halperin [36] presented a semidefinite programming based algorithm with an approximation ratio of $l - (1 - o(1))\frac{l \ln \ln n}{\ln n}$. Note that this condition enforces the doubly exponential bound, $n \geq 2^{2^{l^2}}$, and already for $l = 3$, the hypergraph is very large and hardly suitable for practical purposes.

A further important class consists of hypergraphs with Δ and l being constants. In this case Krivelevich [55] gave an LP-based algorithm that provides an approximation ratio of $l(1 - c\Delta^{\frac{1}{1-l}})$ for some constant $c > 0$. An improved approximation ratio of $l - (1 - o(1))\frac{l(l-1)\ln \ln \Delta}{\ln \Delta}$ was presented by Halperin [36], provided that $l^3 = o(\frac{\ln \ln \Delta}{\ln \ln \ln \Delta})$.

For hypergraphs which are not necessarily uniform, but with edge size bounded from above by a constant l , an improvement of the result of Krivelevich was given by Okun [65]. He proved an approximation ratio of $l(1 - c(\beta, l)\Delta^{-\frac{1}{\beta l}})$ for $\beta \in (0, 1)$ and a constant $c(\beta, l) \in (0, 1)$ depending on β and l , by a modification of the algorithm presented in [55].

2.2 The Randomized Algorithm

A linear programming formulation of hitting set is the following.

$$\begin{aligned}
 \text{(ILP-VC)} \quad & \min \sum_{i=1}^n x_i \\
 & \sum_{i \in E} x_i \geq 1 \text{ for all } E \in \mathcal{E}, \\
 & x_i \in \{0, 1\} \text{ for all } i \in [n] := \{1, \dots, n\}.
 \end{aligned}$$

Its linear programming relaxation, denoted by LP-VC, is given by relaxing the integrality constraints to $x_i \in [0, 1] \forall i \in [n]$. Let Opt

2.2. The Randomized Algorithm

and Opt^* be the value of an optimal solution to ILP-VC and LP-VC, respectively. Clearly, $\text{Opt}^* \leq \text{Opt}$. Let x^* be an optimal solution of LP-VC. Let $\epsilon \in [0, 1]$, we set $\lambda = l(1 - \epsilon)$.

Algorithm 1: VC- \mathcal{H}

Input : A hypergraph $\mathcal{H} = (V, \mathcal{E})$

Output: A hitting set C

1. Initialize $C := \emptyset$.
 2. Solve the LP relaxation of ILP-VC
 3. Set $S_0 := \{i \in [n] \mid x_i^* = 0\}$, $S_1 := \{i \in [n] \mid x_i^* = 1\}$,
 $S_{>} := \{i \in [n] \mid 1 \neq x_i^* \geq \frac{1}{\lambda}\}$ and $S_{<} := \{i \in [n] \mid 0 \neq x_i^* < \frac{1}{\lambda}\}$.
 4. Delete the vertices in S_0 from \mathcal{H} , and set $V := V \setminus S_0$ and
 $\mathcal{E} := \{E \cap V \mid E \in \mathcal{E}\}$.
 5. Take all vertices of S_1 and $S_{>}$ into the hitting set C .
Set $V := V \setminus S_1$ and $\mathcal{E} := \mathcal{E} \setminus \Gamma(S_1)$.
 6. (Randomized Rounding) For all vertices $i \in S_{<}$ include i in the hitting set C , independently for all such i , with probability $x_i^* \lambda$.
 7. (Repairing) Repair the Hitting Set C (if necessary) as follows:
 - a) If $|\{E \in \mathcal{E} \mid E \cap C \neq \emptyset\}| = |\mathcal{E}|$, then return C .
 - b) If $|\{E \in \mathcal{E} \mid E \cap C \neq \emptyset\}| < |\mathcal{E}|$, then pick at most $|\mathcal{E}| - |C|$ additional vertices from arbitrary not covered edges in the hitting set.
 8. Return the hitting set C of \mathcal{H}
-

2. The Hitting Set Problem

Let us briefly explain the ingredients of the algorithm. Usually, as in [24, 26, 30], the LP or semidefinite program is solved and randomized rounding or random hyperplane techniques are used followed by a repairing step. In our algorithm we thin out the hypergraph by removing vertices and edges corresponding to LP-variables with zero value, which will not be taken into the hitting set by randomized rounding (Step 4), *before* entering randomized rounding and repairing. This is an intuitively meaningful sparsening, and in fact will be necessary in Section 2.4 where we estimate the expected size of the repaired hitting set (one step analysis), while in Section 2.3 it is sufficient to analyze randomized rounding and repairing separately.

2.3 Two-Step Analysis of the Algorithm

In this section we analyze the randomized rounding process and greedy repairing separately and then combine the probability estimations (two-step approach). Let X_1, \dots, X_n be 0/1-random variables defined as follows:

$$X_j = \begin{cases} 1 & \text{if the vertex } v_j \text{ was picked into } C \text{ after the rounding step} \\ 0 & \text{otherwise.} \end{cases}$$

Note that the X_1, \dots, X_n are independent. For all $i \in [m]$ we define the 0/1- random variables Z_i as follows

$$Z_i = \begin{cases} 1 & \text{if the edge } E_i \text{ is covered after the rounding step} \\ 0 & \text{otherwise.} \end{cases}$$

2.3. Two-Step Analysis of the Algorithm

Then $Y := \sum_{j=1}^n X_j$ is the cardinality of the hitting set after the randomized rounding step in algorithm VC- \mathcal{H} and $W = \sum_{j=1}^m Z_j$ is the number of covered edges after this step.

For the expected size of the hitting set we have the following upper bound:

$$\mathbb{E}(|C|) \leq \mathbb{E}(Y) + \mathbb{E}(m - W). \quad (2.1)$$

For the computation of the expectation we need the following lemma that gives the exact solution of a constrained optimization problem.

2.1 Lemma. *For all $n \in \mathbb{N}$, $\lambda > 0$ and $x_1, \dots, x_n, z \in [0, 1]$ with $\sum_{i=1}^n x_i \geq z$ and $\lambda x_i < 1$ for all $i \in \mathbb{N}$, we have*

$$\prod_{i=1}^n (1 - \lambda x_i) \leq \left(1 - \lambda \frac{z}{n}\right)^n,$$

and this bound is the tight maximum.

Proof . Let $f(x_1, \dots, x_n) := \prod_{i=1}^n (1 - \lambda x_i)$ and $g(x_1, \dots, x_n) = x_1 + \dots + x_n$, $\lambda \in (0, 1)$. We consider the following maximization problem

$$\begin{aligned} (\text{MAX}) \quad & \max f(x_1, \dots, x_n) \\ & \text{s.t. } g(x_1, \dots, x_n) \geq z \\ & z, x_i \in [0, 1] \quad \forall i = 1, \dots, n \end{aligned}$$

The lemma is proved, if we can show that the maximum of (MAX) is attained for $x_i = z/n$ for all $i = 1, \dots, n$. Let (MAX*) be the problem where we take the equality constraint $g(x_1, \dots, x_n) = z$ in (MAX). Then (MAX*) and (MAX) are equivalent. In fact, if we assume that $g(x_1, \dots, x_n) > z$, then some variable x_i can be decreased and

2. The Hitting Set Problem

in consequence the objective function $f(x_1, \dots, x_n)$ increases. Thus a maximum of (MAX) satisfies $g(x_1, \dots, x_n) = z$. We continue by induction on n . For $n = 1$ the assertion is obviously true. Assume that x_1 has been chosen in an optimal way. Then we have to solve the problem

$$\begin{aligned} (\text{MAX})_{n-1} \quad & \max \frac{f(x_1, \dots, x_n)}{1 - \lambda x_1} \\ & \text{s.t. } g(x_1, \dots, x_n) - x_1 = z - x_1 \\ & x_2, \dots, x_n, z \in [0, 1] \end{aligned}$$

By the induction hypothesis, a solution for $(\text{MAX})_{n-1}$ is

$$x_2 = x_3 = \dots = x_n = \frac{z - x_1}{n - 1}.$$

For this solution we have

$$f(x_1, \dots, x_n) = (1 - \lambda x_1) \left(1 - \lambda \frac{z - x_1}{n - 1} \right)^{n-1} =: h(x_1)$$

Furthermore, the derivative of h is

$$h'(x_1) = \frac{dh(x_1)}{dx_1} = -\lambda \left(1 - \lambda \frac{z - x_1}{n - 1} \right)^{n-1} + (1 - \lambda x_1) \lambda \left(1 - \lambda \frac{z - x_1}{n - 1} \right)^{n-2}.$$

For $x_1^* = z/n$, $h'(x_1^*) = 0$, so $x_1^* = z/n$ is the only extreme point (maximum or minimum) of h in $[0, 1]$.

If for some $a \in [0, 1]$, $h(a) < h(x_1^*)$, then $h(x_1^*)$ must be the global maximum. We show

$$h(0) = \left(1 - \frac{\lambda z}{n - 1} \right)^{n-1} < \left(1 - \frac{\lambda z}{n} \right) \left(1 - \lambda \frac{z - z/n}{n - 1} \right)^{n-1} = h(x_1^*)$$

2.3. Two-Step Analysis of the Algorithm

This inequality is equivalent to

$$\frac{n}{n - \lambda z} < \left(1 + \frac{\lambda z}{n(n - 1 - \lambda z)}\right)^{n-1} \quad (2.2)$$

We prove (2.2) by using the inequality $(1 + a)^n > 1 + an$ for any $n \in \mathbb{N}_{>1}$ and $a > 0$. Note that the assertion of Lemma ?? holds trivially for $n = 1$.

Thus (2.2) is true, if

$$1 + \frac{\lambda z}{n - \lambda z} < 1 + (n-1) \frac{\lambda z}{n(n - 1 - \lambda z)} \left(< \left(1 + \frac{\lambda z}{n(n - 1 - \lambda z)}\right)^{n-1} \right),$$

which is equivalent to

$$\frac{1}{n - \lambda z} < \frac{n - 1}{n} \cdot \frac{1}{n - 1 - \lambda z}. \quad (2.3)$$

Since $\lambda z > 0$, (2.3) holds. Hence $h(0) < h(x_1^*)$ and we proved the lemma. \square

For the analysis of the algorithm we need also the following lemma

2.2 Lemma. *Let $\mathcal{H} = (V, \mathcal{E})$ be a hypergraph with maximum size of edge l and maximum vertex degree Δ , not necessarily constant. Let $\epsilon > 0$.*

- (i) $\mathbb{E}(W) \geq (1 - \epsilon^2)m$.
- (ii) $\text{Opt}^* \geq \frac{m}{\Delta}$.
- (iii) *If $x_j^* > 0$ for all $j \in [n]$, then it holds $\text{Opt}^* \geq \frac{n}{l}$.*
- (iv) *For $\lambda \geq 1$ we have $\text{Opt}^* \leq \mathbb{E}(Y) \leq \lambda \text{Opt}^*$.*

2. The Hitting Set Problem

Proof. (i) Let $i \in [m]$, $|E_i| = r$ and $z_i^* := \sum_{j \in E_i} x_j^* (\geq 1)$. If there is a $j \in E_i$ with $\lambda x_j^* \geq 1$ then $\Pr(Z_i = 0) = 0$, else we have

$$\begin{aligned} \Pr(Z_i = 0) &= \prod_{j \in E_i} (1 - \lambda x_j^*) \stackrel{\text{Lem 2.1}}{\leq} \left(1 - \frac{\lambda z_i^*}{r}\right)^r \\ &\stackrel{z_i^* \geq 1}{\leq} \left(1 - \frac{\lambda}{l}\right)^r = (1 - (1 - \epsilon))^r \\ &\stackrel{r \geq 2}{\leq} \epsilon^2 \end{aligned}$$

and we get

$$\begin{aligned} \mathbb{E}(W) &= \sum_{i=1}^m \Pr(Z_i = 1) = \sum_{i=1}^m (1 - \Pr(Z_i = 0)) \\ &\geq \sum_{i=1}^m (1 - \epsilon^2) \\ &= (1 - \epsilon^2)m. \end{aligned}$$

Note that because of the ILP-VC constraints, the hyperedges with a single vertex are automatically covered by the set S_1 .

(ii) Let $d(v_j)$ the degree of the vertex v_j . With the ILP constraints we have

$$m = \sum_{i=1}^m 1 \leq \sum_{i=1}^m \sum_{j \in E_i} x_j^* = \sum_{j=1}^n d(v_j) x_j^* \leq \Delta \sum_{j=1}^n x_j^* = \Delta \cdot \text{Opt}^*$$

(iii) Let us consider the dual LP for the hitting set LP problem

$$\text{(D-VC)} \quad \max \sum_{j \in \mathcal{E}} y_j$$

2.3. Two-Step Analysis of the Algorithm

$$\begin{aligned} \sum_{j \in \mathcal{E}, i \in j} y_j &\leq 1 && \text{for every } i \in V, \\ y_j &\in [0, 1] && \text{for all } j \in \mathcal{E}. \end{aligned}$$

Let $(y_j^*)_{j \in [m]}$ resp. $\text{Opt}^*(D)$ be an optimal solution of D-VC resp. the value of the optimal solution, then the duality theorem of linear programming applied to (LP-VC) and (D-VC) implies:

- (a) $\text{Opt}^* = \text{Opt}^*(D)$
- (b) If $x_i^* > 0 \Rightarrow \sum_{j \in \mathcal{E}, i \in j} y_j^* = 1$.

Therefore, we have

$$n = \sum_{i \in V} 1 = \sum_{i \in V} \sum_{j \in \mathcal{E}, i \in j} y_j^* = \sum_{j \in \mathcal{E}} y_j^* |j| \leq l \sum_{j \in \mathcal{E}} y_j^* \stackrel{(a)}{=} l \text{Opt}^*.$$

(iv) Let $S \subseteq V$ and set $\text{Opt}^*(S) := \sum_{j \in S} x_j^*$. By the definition of the sets S_1 , $S_>$ and $S_<$, and since every vertex in $S_<$ is picked in the hitting set with probability λx_j^* , we get

$$\begin{aligned} \mathbb{E}(|Y|) &= |S_1| + |S_>| + \sum_{j \in S_<} \lambda x_j^* \\ &\stackrel{\lambda \geq 1}{\geq} |S_1| + \sum_{j \in S_>} 1 + \sum_{j \in S_<} x_j^* \\ &\stackrel{x_j^* \leq 1}{\geq} |S_1| + \sum_{j \in S_>} x_j^* + \sum_{j \in S_<} x_j^* = \text{Opt}^* \end{aligned}$$

On the other hand we have

$$\begin{aligned} \mathbb{E}(|Y|) &= |S_1| + |S_>| + \sum_{j \in S_<} \lambda x_j^* \\ &= |S_1| + \sum_{j \in S_>} 1 + \lambda \text{Opt}^*(S_<), \end{aligned}$$

2. The Hitting Set Problem

since for every $j \in S_>$ we have $\lambda x_j^* \geq 1$, it holds

$$\begin{aligned} \mathbb{E}(|Y|) &\leq |S_1| + \sum_{j \in S_>} \lambda x_j^* + \lambda \text{Opt}^*(S_<) \\ &\stackrel{\lambda \geq 1}{\leq} \lambda |S_1| + \lambda \text{Opt}^*(S_>) + \lambda \text{Opt}^*(S_<) \\ &= \lambda (|S_1| + \text{Opt}^*(S_>) + \text{Opt}^*(S_<)) = \lambda \text{Opt}^*. \end{aligned}$$

□

We proceed to the analysis of the algorithm for hypergraphs with maximum degree and maximum edge size that are not necessarily constant, but are functions of n . The main result for this situation is:

2.3 Theorem. *Let \mathcal{H} be a hypergraph with maximum edge size $l \leq \sqrt{\frac{n}{2}}$ and maximum vertex degree $\Delta \leq \frac{1}{4}n^{\frac{1}{4}}$. The algorithm $VC\text{-}\mathcal{H}$ returns a hitting set C such that $|C| \leq l \left(1 - \frac{1}{8\Delta}\right) \text{Opt}$ with probability at least $\frac{3}{5}$.*

Proof. *Case 1 : $S_0 = \emptyset$.*

Let

$$\epsilon := \frac{l \text{Opt}^*(1 + \beta)}{4m} \quad \text{for} \quad \beta = \frac{\sqrt{2}l}{\sqrt{n}}. \quad (2.4)$$

We can assume that

$$\epsilon \leq \frac{1 + \beta}{4 - \eta}, \quad \text{for all } \eta \in (0, 1), \quad (2.5)$$

because otherwise it follows from the definition of ϵ in (2.4) that $l \text{Opt}^* \geq \frac{4m}{4 - \eta}$, hence $l(1 - \frac{\eta}{4}) \text{Opt}^* \geq m$. Since a hitting set of size m can be trivially found by picking m arbitrary edges and taking one vertex from each of them, pairwise distinct, we can get a $l(1 - \frac{\eta}{4})$ -approximation —i.e. a constant factor strictly better than l — in this

2.3. Two-Step Analysis of the Algorithm

case.

It is straightforward to check that (2.5) implies $\epsilon \leq \frac{2}{3}$, so

$$\lambda = l(1 - \epsilon) \underset{l \geq 3}{>} 1 \quad (2.6)$$

Claim 1. $\Pr\left(W \leq m(1 - \epsilon^2) - \sqrt{\sum_{i=1}^n d^2(v_i)}\right) \leq \frac{1}{5}$.

Proof of Claim 1. First we consider the function: $f(X_1, \dots, X_n) = \sum_{j=1}^m Z_j$. f satisfies:

$$|f(X_1, \dots, X_k, \dots, X_n) - f(X_1, \dots, X'_k, \dots, X_n)| \leq d(v_k),$$

with $X'_k \in \{0, 1\}$ and $X_k \neq X'_k$.

Since the X_1, \dots, X_n are chosen independently at random, by Theorem 1.6 we get for any $t > 0$

$$\Pr(f(X) - \mathbb{E}(f(X)) \leq -t) \leq \exp\left(\frac{-2t^2}{\sum_{i=1}^n d^2(v_i)}\right). \quad (2.7)$$

Let us choose $t = \sqrt{\sum_{i=1}^n d^2(v_i)}$. By Lemma 2.2 (i)

$$\begin{aligned} \Pr\left(W \leq m(1 - \epsilon^2) - \sqrt{\sum_{i=1}^n d^2(v_i)}\right) &\leq \Pr\left(W \leq \mathbb{E}(W) - \sqrt{\sum_{i=1}^n d^2(v_i)}\right) \\ &\underset{\text{Ineq (2.7)}}{\leq} \exp\left(\frac{-2 \sum_{i=1}^n d^2(v_i)}{\sum_{i=1}^n d^2(v_i)}\right) < \frac{1}{5}. \end{aligned}$$

This concludes the proof of Claim 1.

2. The Hitting Set Problem

Claim 2. For $\beta = \frac{\sqrt{2l}}{\sqrt{n}}$ it holds that

$$\Pr(Y \geq l \cdot \text{Opt}^*(1 - \epsilon)(1 + \beta)) < \frac{1}{5}.$$

Proof of Claim 2. The random variables X_1, \dots, X_n in the rounding step are independent. Moreover, since $l \leq \sqrt{\frac{n}{2}}$ we have $\beta \in (0, 1)$. Thus the Angluin-Valliant form of the Chernoff bound ([63], Theorem 2.3 (b), p. 200) shows

$$\begin{aligned} \Pr(Y \geq l(1 - \epsilon)(1 + \beta)\text{Opt}^*) &\stackrel{\text{Lem2.2}(iv)}{\leq} \Pr(Y \geq \mathbb{E}(Y)(1 + \beta)) \\ &\leq \exp\left(-\frac{\beta^2 \mathbb{E}(Y)}{3}\right). \end{aligned}$$

On the other hand we have:

$$\begin{aligned} \frac{\mathbb{E}(Y)\beta^2}{3} &\stackrel{\text{Lem2.2}(iv)}{\geq} \frac{\text{Opt}^*\beta^2}{3} \\ &\stackrel{\text{Lem2.2}(iii)}{\geq} \frac{n\beta^2}{3l} \\ &\geq \frac{2l^2n}{3ln} \\ &\stackrel{l \geq 3}{\geq} 2. \end{aligned}$$

Finally we get:

$$\Pr(Y \geq l(1 - \epsilon)(1 + \beta)\text{Opt}^*) \leq \exp(-2) < \frac{1}{5}.$$

This concludes the proof of Claim 2.

2.3. Two-Step Analysis of the Algorithm

By Claims 1 and 2 we get with probability at least $1 - (\frac{1}{5} + \frac{1}{5}) \geq \frac{3}{5}$ an upper bound for the final hitting set:

$$|C| \leq \underbrace{l(1 - \epsilon)(1 + \beta)\text{Opt}^* + m\epsilon^2}_{(*)} + \underbrace{\sqrt{\sum_{i=1}^n d^2(v_i)}}_{(**)}.$$

By Lemma 2.2(iii) and the condition $\Delta \leq \frac{1}{4}n^{\frac{1}{4}}$ we can continue the estimation:

$$\begin{aligned} (**) &\leq \Delta\sqrt{n} = \sqrt{\frac{n}{l}}\sqrt{l}\Delta \\ &\stackrel{\text{Lem (2.2) (iii)}}{\leq} l\sqrt{\text{Opt}^*}\frac{\Delta^2}{\sqrt{l}}\frac{1}{\Delta} \\ &\leq l\sqrt{\text{Opt}^*}\sqrt{\frac{n}{l}}\frac{1}{16\Delta} \\ &\leq l\text{Opt}^*\frac{1}{16\Delta}. \end{aligned}$$

Furthermore we have

$$\begin{aligned} (*) &\stackrel{\text{Eq (2.4)}}{=} l\left((1 + \beta)(1 - \epsilon) + \frac{l\text{Opt}^*(1 + \beta)^2}{16m}\right)\text{Opt}^* \\ &\stackrel{\text{Lem 2.2(ii)}}{\leq} l(1 + \beta)\left(1 - \frac{3l(1 + \beta)}{16\Delta}\right)\text{Opt}^* \\ &= l\left(1 + \beta - \frac{3l(1 + \beta)^2}{16\Delta}\right)\text{Opt}^*. \end{aligned}$$

On the other hand one can easily check, that

$$\frac{3l(1 + \beta)^2}{16\Delta} - \beta \geq \frac{3}{16\Delta} \tag{2.8}$$

2. The Hitting Set Problem

Let $n \geq 64$. Otherwise we can solve the problem by enumeration. Then we have

$$\begin{aligned} \frac{3l(1+\beta)^2}{16\Delta} - \beta &= \frac{3l(1+\beta)^2 - 16\beta\Delta}{16\Delta} \\ &=_{\beta=\frac{\sqrt{2}l}{\sqrt{n}}} \frac{l}{16\Delta} \left(3(1+\beta)^2 - \frac{16\sqrt{2}\Delta}{\sqrt{n}} \right). \end{aligned}$$

For $n \geq 64$, we have $n^{\frac{1}{4}} \geq 2\sqrt{2}$. By the assumption, $\Delta \leq \frac{1}{4}n^{\frac{1}{4}}$, we get $\Delta \leq \frac{2\sqrt{2}n^{\frac{1}{4}}}{8\sqrt{2}} \leq \frac{\sqrt{n}}{8\sqrt{2}}$. Hence

$$\begin{aligned} \frac{3l(1+\beta)^2}{16\Delta} - \beta &\geq \frac{l}{16\Delta} (3(1+\beta)^2 - 2) \\ &\stackrel{l \geq 3}{\geq} \frac{3}{16\Delta}. \end{aligned}$$

The inequality (2.8) leads to

$$l(1-\epsilon)(1+\beta)\text{Opt}^* + m\epsilon^2 \leq l \left(1 - \frac{3}{16\Delta} \right) \text{Opt}^*.$$

Finally

$$(*) + (**) \leq l \left(1 - \frac{3}{16\Delta} + \frac{1}{16\Delta} \right) \text{Opt}^* = l \left(1 - \frac{1}{8\Delta} \right) \text{Opt}^*.$$

The randomised algorithm returns with probability at least $\frac{3}{5}$ a hitting set C with cardinality at most $l \left(1 - \frac{1}{8\Delta} \right) \text{Opt}^*$.

Case 2: If S_0 is not empty, we can consider the sub-hypergraph \mathcal{H} constructed in step 4 of algorithm VC- \mathcal{H} . Let $\tilde{\Delta}$ resp. \tilde{l} be the maximum vertex degree resp. the maximum edge size of this sub-

2.4. One-Step Analysis of the Algorithm

hypergraph. Now for this hypergraph we have $S_0 = \emptyset$ and with Case 1 we get a hitting set of cardinality at most $\tilde{l}(1 - \frac{\epsilon}{\Delta})\text{Opt}$. Since $\tilde{l} \leq l$ and $\tilde{\Delta} \leq \Delta$, the assertion of Theorem 2.3 holds. \square

Remark 1. For hypergraphs addressed in Theorem 2.3 we have an improvement over the result of Krivelevich [55], for any function $f(n)$ satisfying $f(n) = O(n^{\frac{1}{4}})$, since $n^{\frac{1}{4}} < n^{1-\frac{1}{l}}$ for $l \geq 2$, and our approximation is the better the smaller $f(n)$ becomes. For $\Delta \leq \frac{\ln(n)}{\ln \ln(n)}$ we obtain a better approximation than Halperin [36].

2.4 One-Step Analysis of the Algorithm

Instead bounding the error probability of the randomized rounding step and the repairing step separately as in section 2.3, in this section we analyze the expected size of the hitting set including repairing, and then use concentration inequalities. This approach will lead to new forms of approximations.

For a set $S \subset \{1, \dots, n\}$ let $\text{Opt}^*(S) := \sum_{j \in S} x_j^*$. By (2.1) it holds

$$\mathbb{E}(|C|) \leq \text{Opt}^*(S_1) + l(1 - \epsilon)(\text{Opt}^*(S_{\geq}) + \text{Opt}^*(S_{\leq})) + m\epsilon^2 \quad (2.9)$$

Namely, since every $j \in S_{<}$ is picked with probability $\Pr(j \in C) = \lambda x_j^*$ we have

$$\begin{aligned} \mathbb{E}(|C|) &= |S_1| + |S_{>}| + \sum_{j \in S_{<}} \Pr(j \in C) + m\epsilon^2 \\ &= |S_1| + \sum_{j \in S_{>}} 1 + \sum_{j \in S_{<}} \lambda x_j^* + m\epsilon^2, \end{aligned}$$

2. The Hitting Set Problem

by the definition of $S_{<}$, it holds

$$\begin{aligned}
 \mathbb{E}(|C|) &\leq |S_1| + \sum_{j \in S_{>}} \lambda x_j^* + \sum_{j \in S_{<}} \lambda x_j^* + m\epsilon^2 \\
 &= \text{Opt}^*(S_1) + \lambda (\text{Opt}^*(S_{>}) + \text{Opt}^*(S_{<})) + m\epsilon^2 \\
 &= \text{Opt}^*(S_1) + l(1 - \epsilon) (\text{Opt}^*(S_{>}) + \text{Opt}^*(S_{<})) + m\epsilon^2.
 \end{aligned}$$

We consider the function

$$f: [0, 1] \rightarrow \mathbb{R}, \epsilon \mapsto \text{Opt}^*(S_1) + l(1 - \epsilon)(\text{Opt}^*(S_{>}) + \text{Opt}^*(S_{<})) + m\epsilon^2.$$

f is convex and attains its minimum for

$$\tilde{\epsilon} = \frac{l(\text{Opt}^*(S_{>}) + \text{Opt}^*(S_{<}))}{2m}. \quad (2.10)$$

Moreover we can assume that $\frac{l(\text{Opt}^*(S_{>}) + \text{Opt}^*(S_{<}))}{2m} \in [0, 1]$. Otherwise, if

$$\frac{l(\text{Opt}^*(S_{>}) + \text{Opt}^*(S_{<}))}{2m} > 1$$

then

$$\frac{l}{2} \text{Opt}^* \geq \frac{l}{2} ((\text{Opt}^*(S_{>}) + \text{Opt}^*(S_{<})) > m).$$

Since any hitting set of cardinality m can be found trivially, this approximates the optimum within a factor of $\frac{l}{2} < l$.

Let $S_f := S_{>} \cup S_{<}$. Plugging in $\tilde{\epsilon}$ from (2.10) into (2.9), we get

$$\mathbb{E}(|C|) \leq \text{Opt}^*(S_1) + l \left(1 - \frac{l \text{Opt}^*(S_f)}{4m} \right) \text{Opt}^*(S_f). \quad (2.11)$$

We observe here that the LP-based sparsening of the instance becomes relevant.

2.4. One-Step Analysis of the Algorithm

At next we compute the variance of the size of the hitting set. We get,

2.4 Lemma. *Let X_1, \dots, X_n be the 0/1-random variables returned by algorithm VC- \mathcal{H} . Then we have $\text{Var}(|C|) \leq l\Delta\mathbb{E}(|C|)$.*

Proof. Let Γ and γ like in Lemma 1.7. Furthermore for every $v_i, v_j \in V$, X_i, X_j are dependent iff they belong to the same edge. Thus, for a fixed v_i , there are at the most $(l-1)d(v_i)$ random variables X_j depending on X_i . Furthermore it holds for every $v_i, v_j \in V$:

$$\begin{aligned} \mathbb{E}(X_i X_j) &= \Pr(X_i = 1 \wedge X_j = 1) \\ &\leq \min\{\Pr(X_i = 1), \Pr(X_j = 1)\} \\ &\leq \frac{\Pr(X_i = 1) + \Pr(X_j = 1)}{2}. \end{aligned}$$

Moreover

$$\begin{aligned} \gamma &= \sum_{\{v_i, v_j\} \in \Gamma} \mathbb{E}(X_i X_j) \\ &\leq \sum_{\{v_i, v_j\} \in \Gamma} \frac{\Pr(X_i = 1) + \Pr(X_j = 1)}{2} \\ &\leq \sum_{i=1}^n \frac{(l-1)d(v_i)}{2} \Pr(X_i = 1) \\ &= \frac{(l-1)d(v_i)}{2} \sum_{i=1}^n \Pr(X_i = 1) \\ &= \frac{(l-1)\Delta}{2} \mathbb{E}(|C|) \end{aligned}$$

2. The Hitting Set Problem

so with Lemma 1.7

$$\begin{aligned}\text{Var}(|C|) &\leq \mathbb{E}(|C|) + 2\gamma \\ &\leq \mathbb{E}(|C|) + (l-1)d(v_i)\mathbb{E}(|C|) \\ &= ((l-1)d(v_i) + 1)\mathbb{E}(|C|) \\ &\leq l\Delta\mathbb{E}(|C|).\end{aligned}$$

□

Quasi-Regularisable l -Uniform Hypergraphs. First we give the definition of the Quasi-Regularisable hypergraphs

2.5 Definition. For an integer $k \geq 0$, multiplying the edge E_i by k means replacing the edge E_i in \mathcal{H} by k identical copies of E_i . If $k = 0$, this operation is the deletion of the edge E_i . A hypergraph \mathcal{H} is called *regularisable* if a regular hypergraph can be obtained from \mathcal{H} by multiplying each edge E_i by an integer $k_i \geq 1$. Finally, a hypergraph \mathcal{H} is called *quasi-regularisable* if a regular hypergraph is obtained by multiplying each edge e_i by an integer $k_i \geq 0$ where $\sum_i^m k_i > 0$. Regular implies regularisable and this implies quasi-regularisable (see [10]). Note that quasi-regularisable hypergraphs play an important role in the study of matching and covering in hypergraphs, e. g., [27].

Recall that S_1 is the set $S_1 = \{j \in [n] \mid x_j^* = 1\}$, containing those vertices for which the LP-optimal solution is tight (see algorithm VC- \mathcal{H} , step 3).

The next theorem is the main result of this section and it is proved using the above stated estimation (2.11) of $\mathbb{E}(|C|)$ and the Chebychev-Cantelli inequality.

2.4. One-Step Analysis of the Algorithm

2.6 Theorem. *Let \mathcal{H} be a l -uniform, quasi-regularisable hypergraph with arbitrary l and maximum vertex degree $\Delta = O(n^{\frac{1}{3}})$, then the algorithm $VC\text{-}\mathcal{H}$ returns a hitting set C such that*

$$|C| \leq l \left(1 - \frac{n}{8m}\right) \text{Opt}^*$$

with probability at least $\frac{3}{4}$.

We need the following theorem of Berge [10].

2.7 Theorem. *For an l -uniform hypergraph \mathcal{H} , the following properties are equivalent:*

1. \mathcal{H} is quasi-regularisable;
2. $\text{Opt}^* = \frac{n}{l}$ (i. e., the vector $x^* = (\frac{1}{l}, \dots, \frac{1}{l})$ is an optimal solution for the LP relaxation and l is the size of the edges).

By this theorem, the condition $S_1 = \emptyset$ has a graph-theoretical meaning.

Proof of Theorem 2.6. By (2.11) and Theorem 2.7 we get for quasi-regularisable l -uniform hypergraphs with arbitrary l and bounded degree Δ the approximation

$$\mathbb{E}(|C|) \leq l \left(1 - \frac{n}{4m}\right) \text{Opt}^*. \quad (2.12)$$

Hence

$$\begin{aligned} \Pr \left(|C| \geq l \left(1 - \frac{n}{8m}\right) \text{Opt}^* \right) &= \Pr \left(|C| \geq l \left(1 - \frac{n}{4m}\right) \text{Opt}^* + \frac{nl\text{Opt}^*}{8m} \right) \\ &\stackrel{\text{Th 2.9}}{\leq} \Pr \left(|C| \geq \mathbb{E}(|C|) + \frac{nl\text{Opt}^*}{8m} \right) \end{aligned}$$

2. The Hitting Set Problem

$$\stackrel{\text{Th 1.5}}{\leq} \frac{1}{1 + \frac{\left(\frac{n \text{Opt}^*}{8m}\right)^2}{\text{Var}(|C|)}}$$

For $n \geq 8^2 \Delta^3$ we get with a straightforward calculation that

$$\frac{\left(\frac{\ln \text{Opt}^*}{8m}\right)^2}{\text{Var}(|C|)} \geq l \geq 3.$$

So we obtain a hitting set C of size at most $l\left(1 - \frac{n}{8m}\right) \text{Opt}^*$ with probability at least $\frac{3}{4}$. \square

Remark 2. In Theorem 2.6, we can assume that $n < 8m$, because otherwise we have $\text{Opt}^* = \frac{n}{l} \geq \frac{8m}{l}$ thus $m \leq \frac{l}{8} \text{Opt}^*$. By taking one vertex for each edge we obtain a hitting set of cardinality $\frac{l}{8} \text{Opt}^*$, which gives an approximation ratio of $l/8$. For hypergraphs addressed in Theorem 2.6 we have an improvement over the ratio of Krivelevich if $m \leq cn^{\frac{2l-1}{l}}$ and the ratio of Halperin if $m \leq \frac{(1-o(1))^{-1} \ln(n)n}{\ln \ln(n)}$.

l -Uniform Hypergraphs with Bounded Vertex Degree.

In this section l and Δ are constants and \mathcal{H} is an l -uniform hypergraph. Let $\tilde{\mathcal{H}} = (\tilde{V}, \tilde{\mathcal{E}})$ be the sub-hypergraph of \mathcal{H} constructed in step 5 of the algorithm VC- \mathcal{H} with $|\tilde{V}| = \tilde{n}$ and $|\tilde{\mathcal{E}}| = \tilde{m}$. We denote by \tilde{l} and $\tilde{\Delta}$ the maximum size of all edges and the maximum vertex degree in $\tilde{\mathcal{H}}$. We consider the LP relaxation of the ILP formulation of the hitting set problem in $\tilde{\mathcal{H}}$ which we denote by $\text{LP}(\tilde{\mathcal{H}})$. By $\text{Opt}^*(\tilde{\mathcal{H}})$ we denote the value of the optimal solution of $\text{LP}(\tilde{\mathcal{H}})$. The optimal LP solution for \mathcal{H} is Opt^* . Then the following holds.

2.8 Lemma. $\text{Opt}^*(\tilde{\mathcal{H}}) = \text{Opt}^* - |S_1|$ and $\mathbb{E}(|C|) \leq |S_1| + \mathbb{E}(|\tilde{C}|)$.

2.4. One-Step Analysis of the Algorithm

2.9 Lemma. *Let l and Δ be constants, and let \mathcal{H} be a l -uniform hypergraph with maximum vertex degree Δ such that $3 \leq l \leq \frac{8}{3}\Delta$. Then: $\mathbb{E}(|C|) \leq l \left(1 - \frac{l}{4\Delta}\right) \text{Opt}^*$.*

Proof . Since there are no 1's in the solution $(\tilde{x}_1, \dots, \tilde{x}_{\tilde{n}})$, there is no tight LP($\tilde{\mathcal{H}}$)-variable. By plugging $\tilde{\epsilon}$ in (2.9) for $\tilde{\mathcal{H}}$, we get

$$\begin{aligned} \mathbb{E}(|\tilde{C}|) &\leq l \left(1 - \frac{l\text{Opt}^*(\tilde{\mathcal{H}})}{2m}\right) \text{Opt}^*(\tilde{\mathcal{H}}) + \tilde{m} \left(\frac{l\text{Opt}^*(\tilde{\mathcal{H}})}{2m}\right)^2 \\ &\stackrel{m \geq \tilde{m}}{\leq} l \left(1 - \frac{l\text{Opt}^*(\tilde{\mathcal{H}})}{2m}\right) \text{Opt}^*(\tilde{\mathcal{H}}) + m \left(\frac{l\text{Opt}^*(\tilde{\mathcal{H}})}{2m}\right)^2 \\ &= l \left(1 - \frac{l\text{Opt}^*(\tilde{\mathcal{H}})}{4m}\right) \text{Opt}^*(\tilde{\mathcal{H}}). \end{aligned}$$

Let $\bar{\lambda} := l \left(1 - \frac{l}{4\Delta}\right)$. It is straightforward to check that for $3 \leq l \leq \frac{8}{3}\Delta$, $\bar{\lambda} \geq 1$. Thus

$$\begin{aligned} \mathbb{E}(|C|) &= |S_1| + \mathbb{E}(|\tilde{C}|) \\ &\leq |S_1| + l \left(1 - \frac{l\text{Opt}^*(\tilde{\mathcal{H}})}{4m}\right) \text{Opt}^*(\tilde{\mathcal{H}}) \\ &\stackrel{\bar{\lambda} \geq 1}{\leq} l \left(1 - \frac{l}{4\Delta}\right) |S_1| + l \left(1 - \frac{l\text{Opt}^*(\tilde{\mathcal{H}})}{4m}\right) \text{Opt}^*(\tilde{\mathcal{H}}) \\ &\stackrel{m \geq \tilde{m}, \text{Lem 2.2 (ii)}}{\leq} l \left(1 - \frac{l}{4\Delta}\right) |S_1| + l \left(1 - \frac{l}{4\bar{\Delta}}\right) \text{Opt}^*(\tilde{\mathcal{H}}). \end{aligned}$$

Because $\Delta \geq \tilde{\Delta}$, we have: $\mathbb{E}(|C|) \leq l \left(1 - \frac{l}{4\Delta}\right) \text{Opt}^*$. □

Lemma 2.9 and Lemma 2.4 imply the following theorem using the Chebyshev-Cantelli inequality and standard calculations.

2. The Hitting Set Problem

2.10 Theorem. *Let \mathcal{H} be an l -uniform hypergraph with bounded vertex degree with $3 \leq l \leq \frac{8}{3}\Delta$, then the algorithm $VC\text{-}\mathcal{H}$ returns a hitting set C such that*

$$|C| \leq l \left(1 - \frac{l-1}{4\Delta}\right) \text{Opt}^*$$

with probability at least $\frac{3}{4}$.

Proof. W.l.o.g. we may assume that $m \geq 16\Delta^5$. Otherwise, we can solve the problem by enumeration. We have

$$\begin{aligned} \Pr\left(|C| \geq l \left(1 - \frac{l-1}{4\Delta}\right) \text{Opt}^*\right) &= \Pr\left(|C| \geq l \left(1 - \frac{l}{4\Delta}\right) \text{Opt}^* + \frac{l\text{Opt}^*}{4\Delta}\right) \\ &\stackrel{\text{Lem 2.9}}{\leq} \Pr\left(|C| \geq \mathbb{E}(|C|) + \frac{l\text{Opt}^*}{4\Delta}\right) \\ &\stackrel{\text{Th 1.5}}{\leq} \frac{1}{1 + \frac{\left(\frac{l\text{Opt}^*}{4\Delta}\right)^2}{\text{Var}(|C|)}}. \end{aligned}$$

And $m \geq 16\Delta^5$ may continue,

$$\begin{aligned} \frac{\left(\frac{l\text{Opt}^*}{4\Delta}\right)^2}{\text{Var}(|C|)} &\stackrel{\text{Lem 2.4}}{\geq} \left(l \frac{(\text{Opt}^*)^2}{16\Delta^3 \mathbb{E}(|C|)}\right) \\ &\stackrel{\mathbb{E}(|C|) \geq l\text{Opt}^*}{\geq} \frac{\text{Opt}^*}{16\Delta^3} \\ &\stackrel{\text{Lem 2.2}(iii)}{\geq} \frac{\Delta m}{16\Delta^5} \geq \Delta. \end{aligned}$$

Therefore we get

$$\Pr\left(|C| \geq l \left(1 - \frac{l-1}{4\Delta}\right) \text{Opt}^*\right) \leq \frac{1}{1 + \Delta} \stackrel{\Delta \geq 3}{\leq} \frac{1}{4}.$$

2.5. Partial Vertex Cover in Hypergraphs

Hence the theorem holds. \square

This improves over the result of Krivelevich [55] for Δ smaller than $(l-1)^{1+\frac{1}{l-2}}$ and of Okun [65] for Δ smaller than $(l-1)^{1+\frac{1}{\beta l-1}}$. The approximation ratio in this result is little weaker than the ratio of Halperin [36]. But the advantage here is that l and Δ are not coupled anymore, so a significantly larger class of hypergraphs than in [36] is covered.

2.5 Partial Vertex Cover in Hypergraphs

The k -partial vertex cover in hypergraphs is a generalization of the hitting set problem.

A set $X \subseteq V$ is called k -partial vertex cover for \mathcal{H} if at least k edges of \mathcal{H} are incident in X . The (unweighted) k -partial vertex cover problem for hypergraphs is to find a k -vertex cover of minimum cardinality. If k is equal to the number of hyperedges, we have the hitting set problem (or vertex cover problem) in hypergraphs. For graphs it is the classical vertex cover problem in combinatorial optimization, whose approximation complexity has been studied for nearly 4 decades. The partial vertex cover problem in graphs is known to be \mathcal{NP} -hard and it has been a long standing open problem to beat the 2-approximation. For general graphs, the first 2-approximation algorithm for the partial vertex cover problem was given by Bshouty and Burroughs [12]. A faster approximation algorithm achieving the same factor 2 was later given by Hochbaum [39] and Bar-Yehuda [9].

For graphs with maximum vertex degree at most Δ , Gandhi, Khuller and Srinivasan [30] gave the first algorithm with approximation ratio smaller than 2. They presented an algorithm based on the randomized

2. The Hitting Set Problem

rounding procedure that achieves a factor of $2 \left(1 - \Omega\left(\frac{1}{\Delta}\right)\right)$. Improvements have been obtained by Halperin and Srinivasan [35].

For hypergraphs the partial vertex cover problem was first studied by Kearns [49] who gave an approximation ratio of $O(H(n))$. For hypergraphs with bounded degree Δ , Slavák [72] achieved an approximation ratio of $O(H(\Delta))$. For hypergraphs with edge size at most l , Bar-Yehuda [9] gave an algorithm based on the local-ratio method with approximation guarantee l . Later, Gandhi, Khuller and Srinivasan [30] achieved the same approximation ratio using a primal/dual approach. But the problem of breaking the l -ratio remained open. In this section we will resolve this problem.

An integer linear programming formulation of the k -partial vertex cover in \mathcal{H} is the following:

$$\begin{aligned}
 (\text{ILP} - k - \text{VC}) \quad & \min \sum_{i=1}^n x_i \\
 & \sum_{i \in j} x_i \geq z_j \quad \text{for all } j \in [m] := \{1, \dots, m\} \\
 & \sum_{j=1}^m z_j \geq k \\
 & x_i, z_j \in \{0, 1\} \quad \text{for all } j \in [m], i \in [n].
 \end{aligned}$$

We apply the randomized algorithm VC- \mathcal{H} without step 3 to the partial vertex cover problem in hypergraphs. We will show that it achieves approximation ratios below l for hypergraphs with hyperedge degree at most a constant D and for hypergraphs with vertex degree at most a constant Δ .

We consider hypergraphs with hyperedge size at most l , $l \geq 3$, not necessarily constant. In the case that the maximum hyperedge degree

2.5. Partial Vertex Cover in Hypergraphs

is at most a constant D we have

2.11 Theorem. *Let \mathcal{H} be a hypergraph with edge size at most l , $l \in \mathbb{N}, l \geq 3$, and hyperedge degree at most a constant D , $D \in \mathbb{N}$. The algorithm k -VC returns a k -partial vertex cover C such that*

$$|C| \leq l \left(1 - \Omega \left(\frac{1}{D+1} \right) \right) \text{Opt} \quad \text{with probability at least } \frac{3}{5}.$$

In the case that the vertex degree is at most a constant Δ , we will prove

2.12 Theorem. *Let \mathcal{H} be a hypergraph with hyperedge size at most l , $l \in \mathbb{N}, l \geq 3$, and vertex degree at most a constant Δ , $\Delta \in \mathbb{N}$. For $k \geq \frac{m}{4}$ the algorithm k -VC returns a k -partial vertex cover C such that*

$$|C| \leq l \left(1 - \Omega \left(\frac{1}{\Delta} \right) \right) \text{Opt} \quad \text{with probability at least } \frac{3}{5}.$$

To our best knowledge, the presented results for the partial vertex cover problem in hypergraphs are the first that break the barrier of l . Both theorems are proved in the same manner as Theorem 2.3. Namely, as above, we consider two kinds of $\{0, 1\}$ -random variables. We define X_1, \dots, X_n by

$$X_i = \begin{cases} 1 & \text{if } v_i \text{ was picked into the cover after the rounding step} \\ 0 & \text{otherwise.} \end{cases}$$

Note that the X_1, \dots, X_n are independent. For all $j \in [m]$ we define

2. The Hitting Set Problem

the $\{0, 1\}$ - random variables Z_j as follows

$$Z_j = \begin{cases} 1 & \text{if the hyperedge } E_j \text{ is covered after the rounding step} \\ 0 & \text{otherwise.} \end{cases}$$

Then $X := \sum_{i=1}^n X_i$ is the cardinality of the cover after the randomised rounding step in the algorithm and $Z = \sum_{j=1}^m Z_j$ is the number of covered hyperedges after this step. For the expected size of the cover we have the following upper bound:

$$\mathbb{E}(|C|) \leq \mathbb{E}(X) + \mathbb{E}(\max\{k - Z, 0\}) \quad (2.13)$$

For the computation of the expectation of X and Z we get with the same argument as in Lemma (2.2):

$$\mathbb{E}(Z) \geq (1 - \epsilon^2)k,$$

$$\text{Opt}^* \leq \mathbb{E}(X) \leq \lambda \text{Opt}^*.$$

Hence (2.13) becomes

$$\mathbb{E}(|C|) \leq \lambda \text{Opt}^* + \epsilon^2 k. \quad (2.14)$$

The idea of the proof is to find a suitable ϵ that satisfies the Equation (2.14) and separately estimate the size of the covers X and $k - Z$ returned by the randomized algorithm and greedy repair, respectively. This will be obtained by the following claim,

Claim 1. For $\epsilon := \frac{\text{Opt}^*(1+\beta)}{k}$ with $\beta = \frac{1}{3(D+1)}$ we have

$$(i) \Pr(X \geq l \cdot \text{Opt}^*(1 - \epsilon)(1 + \beta)) < \frac{1}{5}.$$

$$(ii) \Pr\left(W \leq k(1 - \epsilon^2) - 2\sqrt{k(D+1)}\right) \leq \frac{1}{5}.$$

2.6. Summary and Further Work

To prove assertion (i) we use the Chernoff bound, since the random variables returned by the randomized rounding step are independent. For the assertion (ii) we cannot use the Chernoff bound because of the dependency of the random variables returned by the greedy repairing step are no longer independent. We use the Chebychev-Cantelli inequality. Finally, by the union bound one can finish the proof of Theorem 2.11.

□

The proof of Theorem 2.12 is similar to the proof of Theorem 2.3. Note that the results for the hitting set problem are better than those for the partial vertex cover problem. For more details see [20].

2.6 Summary and Further Work

In chapter 2 we discussed the hitting set problem and its generalization, the partial vertex cover problem in hypergraphs. We proposed and analyzed an approximation algorithm for both problems on subclasses of hypergraphs that are mainly governed by three parameters - maximal edge size l , maximum vertex degree Δ and the maximum edge degree D .

The presented randomized algorithm combines LP-based randomized rounding, sparsening of the hypergraph and greedy repairing. Such a hybrid approach is frequently used in practice and it has been analyzed for many problems, e. g., maximum graph bisection [26], maximum graph partitioning problems [24, 44] and the (full) vertex cover problem and partial vertex cover problem in graphs [30, 36]. We analyzed the algorithm in two different ways that lead to three new results:

2. The Hitting Set Problem

For the hitting set problem, we considered hypergraphs with maximum edge size l and maximum vertex degree Δ , for the moment not necessarily assumed to be constants.

We have shown that our algorithm achieves for $l = O(\sqrt{n})$ and $\Delta = O(n^{\frac{1}{4}})$ an approximation ratio of $l(1 - \frac{c}{\Delta})$, for some constant $c > 0$, with constant probability. In this case, our result improves the result of Krivelevich, for any function $f(n)$ satisfying $f(n) = O(n^{\frac{1}{4}})$, since $n^{\frac{1}{4}} < n^{1-\frac{1}{l}}$ for $l \geq 2$, and the approximation is the better the smaller $f(n)$ becomes. For $\Delta \leq \frac{\ln n}{\ln \ln n}$ we obtain a better approximation than Halperin. In Section 2.4.1 we analyzed the algorithm for the class of uniform, quasi-regularisable hypergraphs, which are known and useful in the combinatorics of hypergraphs (see Berge [10]). We proved an approximation ratio of $l(1 - \frac{n}{8m})$ provided that $\Delta = O(n^{\frac{1}{3}})$. This result improves the approximation ratio given by Krivelevich and Halperin for sparse hypergraphs (where, roughly speaking, sparseness means $m \leq n^\alpha$, $\alpha \leq 2$). In Section 2.4.2 we considered l -uniform hypergraphs, where l and Δ are constants, and achieved a ratio of $l(1 - \frac{l-1}{4\Delta})$. This improves the result of Krivelevich for Δ smaller than $(l-1)^{1+\frac{1}{l-2}}$ and of Okun for Δ smaller than $(l-1)^{1+\frac{1}{\beta l-1}}$, respectively.

For the partial vertex cover problem we considered hypergraphs with constant maximum edge degree D and maximum edge size l not necessarily assumed to be constant and yielded an approximation ratio of $l(1 - \Omega(1/(D+1)))$. To our knowledge this result is the first one that breaks the barrier of l .

We believe that the analysis presented in this paper can incorporate other hypergraph parameters in a natural way, like bounded VC-dimension, uncrowdedness, or exclusion of subhypergraphs. We hope

2.6. Summary and Further Work

that this may lead to new and better approximation results for the hitting set problem in such hypergraphs. Another challenge is the derandomization of our hybrid algorithm.

Inapproximability of \mathbf{b} -Matching in l -Uniform Hypergraphs

Let $\mathbf{b} \in \mathbb{N}^n$ where for every $i \in V$, $b_i = b$ (a positive integer). The \mathbf{b} -matching problem is a prototype of packing integer programs. It has been studied intensively in combinatorics and optimization and has several applications, for example in medicine [25] and combinatorial auctions [31].

In the following we study the complexity of the maximum cardinality \mathbf{b} -matching problem in l -uniform hypergraphs with focus on its approximability.

An outline of this chapter¹ is as follows. In Section 3.1 we present the problem, cite some related works and previous results and then we introduce definitions and useful tools. In Section 3.3 we state a fundamental existence theorem that we call the Structure Theorem and give its proof. Section 3.4 contains the reduction from an instance of the problem *Max-E3-Lin-q* problem to an instance of the \mathbf{b} -matching problem and we prove the inapproximability result. We finally briefly discuss the main topics of the chapter in section 3.5 and give an overview of future works.

¹This chapter is mainly based on the paper [19]

3. Inapproximability of \mathbf{b} -Matching in l -Uniform Hypergraphs

3.1 Problem and Related Work.

For the readers convenience we briefly recall the definition of the \mathbf{b} -matching problem in hypergraphs.

Let $\mathcal{H} = (V, \mathcal{E})$ be a hypergraph, where $\mathcal{E} = \{e_1, e_2, \dots, e_m\}$. Let $w : \mathcal{E} \rightarrow [0, 1] \cap \mathbb{Q}$ be a weight function on the hyperedges. We call $w(e)$ the weight of an hyperedge $e \in \mathcal{E}$. For a given $b \in \mathbb{N}$, the \mathbf{b} -matching problem in \mathcal{H} is task of finding a subset $M \subseteq \mathcal{E}$, such that no vertex is contained in more than b hyperedges of M and with maximum weight, .i.e, $w(M) = \max \sum_{j=1}^m w(e_j)$.

The \mathbf{b} -matching problem in hypergraphs is \mathcal{NP} -hard, even for $l = 3$ [48] [59]. Its integer linear programs can be written in the form:

$$\begin{aligned} & \max \sum_{j=1}^m w_j x_j \\ & \sum_{j \in [m]} a_{ij} x_j \leq b \quad \text{for all } i \in [n], \\ & x_j \in \{0, 1\} \text{ for all } j \in [m]. \end{aligned}$$

Here, $a_{ij} \in \{0, 1\}$ is the vertex-edge incidence matrix of the hypergraphs, that is $a_{ij} = 1$ if and only if vertex i is contained in hyperedge j ; and $x_j = 1$ if and only if hyperedge j is chosen for the \mathbf{b} -matching. The \mathbf{b} -matching problem in hypergraphs is a prominent problem in combinatorics and optimization, and on its algorithmic and complexity aspect has been intensively explored.

Approximability of the Problem

\mathbf{b} -Matching in Graphs. In the case $\mathbf{b} = \mathbf{1}$ ($\mathbf{1} = (1, 1, \dots, 1)$), the problem is solvable in polynomial time [18]. The same holds also if

3.1. Problem and Related Work.

$b_v \geq 1$ for all $v \in V$, by extending Edmonds algorithm for matching in graphs (see Marsh [62]).

b-Matching in Hypergraphs. For $\mathbf{b} = \mathbf{1}$, the problem is known also as the set packing problem. The approximability of the problem has been intensively studied under various aspects (See [4, 28, 40, 43, 61, 68, 71, 73]). For the weighted problem, the currently best known algorithm was presented by Berman [11]. He proved that the problem can be approximated within $\frac{l+1+\epsilon}{2}$ for any $\epsilon > 0$, where l is the maximum hyperedge size. For the unweighted case Hurkens and Schrijver [43] showed that the problem is approximable within $\frac{l}{2} + \epsilon$ for any $\epsilon > 0$.

In case of $\mathbf{b} = (b, b, \dots, b)$ for $b \in \mathbb{N}_{\geq 1}$ several results are known. The most important of these are based on the randomized rounding technique.

A randomized algorithm for \mathbf{b} -matching in hypergraphs was first proposed by Raghavan and Thompson [68]. The analysis of their algorithm relies on the application of Chernoff bounds to sums of independent Bernoulli random variables [13]. Later, using the same technique, Srivastav and Stangier [76] extended the result from the unweighted problem to the weighted problem. Both works are based on the assumption that $b = \Omega(\ln n)$. For \mathbf{b} -matching problems with arbitrary vector $\mathbf{b} = (b_1, b_2, \dots, b_n)$ the best approximation ratio of $O\left(\min\left\{n^{\frac{1}{b_{\min}+1}}, l^{\frac{1}{b_{\min}}}\right\}\right)$ where

$b_{\min} = \min_{v \in V} b_v$, was obtained by Srinivasan [74] who used the randomized rounding technique as in [76]. For the derandomization of the algorithm Srinivasan used positive correlation of the PIPs which allowed the application of the FKG inequality and leads to an improved pessimistic estimator. Recently, Krysta [56] presented an

3. Inapproximability of \mathbf{b} -Matching in l -Uniform Hypergraphs

analysis of a primal-dual based greedy approach that achieves a ratio of $O(l)$.

There is a large body of work on the non-approximability of the 1-matching problem. Håstad [33] proved that the 1-matching problem cannot be approximated within any ratio of $O(n^{1-\epsilon})$ unless $\mathcal{NP} \subseteq \mathcal{ZPP}$. Three years later Gonen and Lehmann [31] showed by reduction from the clique problem that no polynomial-time approximation better than a ratio of \sqrt{n} is possible unless $\mathcal{NP} = \mathcal{ZPP}^2$. Hazan, Safra and Schwartz [37] proved for l -uniform hypergraphs that there is no polynomial-time algorithm that approximates the 1-matching problem to within any ratio smaller than $O(\frac{l}{\ln l})$, unless $\mathcal{P} = \mathcal{NP}$, improving the result of Trevisan [77], who showed that l -independent set cannot be approximated to within $\frac{l}{2^{O(\sqrt{\ln l})}}$. But to our knowledge, no such results are known for the \mathbf{b} -matching problem for $b \geq 2$ and l -uniform hypergraphs. Recent work of Bansal and Khot [53] to prove non-approximability results apply to the vertex cover problem in hypergraphs and 2-variable problems, like max-cut, but to our knowledge its impact to the \mathbf{b} -matching or general packing problems has not been explored yet.

3.2 Definitions and complexity-theoretic Tools

In this section we state some special definitions and theorems needed in this chapter.

Many non-approximability results have been derived via the so-called gap technique. This means a \mathcal{NP} -hard reduction will be used to

² \mathcal{ZPP} is the class of problems for which there exists a randomized algorithm with polynomially runtime and that makes no failure (i.e., such an algorithm either provides the correct result or fails by producing the output

3.2. Definitions and complexity-theoretic Tools

create a gap in the objective function of the constructed instances. We start with the definition of a gap problem:

3.1 Definition. Let P be an optimization problem with cost function w and let S be the set of all feasible solutions of P and $a, b \in \mathbb{R}$ with $a < b$. Then the (a, b) -gap-problem of P is the following decision problem:

For a given instance I of P decide, whether

- (i) there exists an $s \in S(I)$ with $w(I, s) \geq b$, or *(Completeness)*
- (ii) every $s \in S(I)$ satisfies $w(I, s) \leq a$ *(Soundness)*

3.2 Corollary. Let P be an optimization problem such that its (a, b) -gap-problem is \mathcal{NP} -hard. Then it is \mathcal{NP} -hard to approximate P within any ratio smaller than $\frac{b}{a}$.

Proof: Let P be a maximization problem and assume for a moment that there exists a polynomial time approximation algorithm A such that

$$A(I) > \frac{a}{b} \text{Opt}(I) \quad \text{for all instances } I \text{ of } P.$$

Let I be an instance of P and $s \in S(I)$ a feasible solution with $A(I) = w(I, s)$. We differ between two cases:

1. I is an instance of the first type, i. e., the value of each solution is at most a .

Then we have also $\text{Opt}(I) \leq a$ and we have

$$A(I) \leq \text{Opt}(I) \leq a.$$

The algorithm A returns in this case a solution of value at most a

3. Inapproximability of \mathbf{b} -Matching in l -Uniform Hypergraphs

2. There exists a solution for I of value at least b .

Then it holds $\text{Opt}(I) \geq b$ and we have:

$$A(I) > \frac{a}{b} \text{Opt}(I) \geq a.$$

In this case the algorithm A returns a solution of value strictly greater than a . With the gap property it follows that it must be a solution of value at least b . So the algorithm solves the decision problem (a, b) -gap- P in polynomial time, in contradiction to its \mathcal{NP} -hardness. \square

The aim is to prove for some adequate $a, b \in \mathbb{Q}$ that the (a, b) -gap problem corresponding to the \mathbf{b} -matching problem in uniform hypergraphs is \mathcal{NP} -hard. Therefore we have to transform an instance of a suitable \mathcal{NP} -hard optimization problem P to an instance of our treated problem and to map the gap of P to a gap of the \mathbf{b} -matching.

Let us now consider the *Max-E3-Lin- q* problem.

3.3 Definition. Let $q \in \mathbb{N}$. *Max-E3-Lin- q* problem is the following optimization problem:

Input: A system Φ of linear equations over \mathbb{Z}_q with exactly 3 (distinct) variables in each equation.

Goal: Find an assignment that satisfies the maximal number of equations of Φ .

The following fundamental theorem is due to Håstad [33].

3.4 Theorem. *For every $q \in \mathbb{N}$ and $\epsilon > 0$ $(\frac{1}{q} + \epsilon, 1 - \epsilon)$ -gap-problem of *Max-E3-Lin- q* is \mathcal{NP} -hard. Furthermore the result holds for instances*

3.3. The Construction and the Inapproximability Result.

of *Max-E3-Lin- q* in which each variable occurs exactly a constant number of times.

The following definitions are specially adapted to the **b**-matching problem in hypergraphs.

3.5 Definition. (Concentrated (k, δ) -Partition **b-Matching)** Let $\mathcal{H} = (V, \mathcal{E})$ be an l -uniform hypergraph. We say that \mathcal{H} possesses the concentrated (k, δ) -partition **b**-matching property, if there exists a partition $\mathcal{E}_1, \dots, \mathcal{E}_k$ of \mathcal{E} with $|\mathcal{E}_1| = \dots = |\mathcal{E}_k|$ such that for every **b**-matching M in \mathcal{H} there is an $i \in [k]$, such that

$$|M \setminus \mathcal{E}_i| \leq \delta |\mathcal{E}|.$$

In the rest of this chapter we will abbreviate the term concentrated (k, δ) -partition **b**-matching to: (k, δ) -(P.b-MC) property.

3.6 Definition. ((q, b) -hyperedge coloring.) Put $[q] := \{1, \dots, q\}$. We say a hypergraph $\mathcal{H} = (V, \mathcal{E})$ is (q, b) -hyperedge colorable, if there exists a coloring function $f : \mathcal{E} \rightarrow [q]$ such that every vertex is contained in at most b hyperedges of the same color.

In the next section we give a polynomial-time transformation from *Max-E3-Lin- q* to the **b**-matching problem for l -uniform hypergraphs.

3.3 The Construction and the Inapproximability Result.

The reduction presented in this section is based on the existence of a class of hypergraphs with particular properties. The theorem presented below shows under which conditions such hypergraphs exists.

3. Inapproximability of \mathbf{b} -Matching in l -Uniform Hypergraphs

3.7 Theorem. (Structure Theorem) *For every $q > 1$, for every $b \geq 1$ there exists a constant $c(q, b)$ ($c(q, b)$ may depend on q and b) and for every $t > c(q, b)$ there exists a hypergraph $\mathcal{H} = (V, \mathcal{E}(\mathcal{H}))$ such that the following holds:*

1. $V = [t] \times [d]$, where $d = \Theta(q \ln q)$.
2. \mathcal{H} has the concentrated $(q, \frac{1}{q^2})$ -partition b -matching property.
3. \mathcal{H} is d -uniform.
4. \mathcal{H} is bq -regular.
5. \mathcal{H} is (q, b) -hyperedge colorable.

Because of the complexity of the proof of the structure theorem and for the convenience of the reader, we prefer to state it in the last section (Section 3.4) of the chapter.

Let us denote a hypergraph satisfying the properties of Theorem 3.7 by $H_{b,q}$. Due to the uniformity, regularity and (q, b) -hyperedge colorability of $H_{b,q}$, its set of hyperedges is of cardinality bqt and it may be partitioned corresponding to the (q, b) -hyperedge colorability into q color sets where every vertex in $H_{b,q}$ is contained in exactly b hyperedges of the same color.

a) The Construction of the Instance.

In the current section we transform in polynomial time an instance of *Max-3-LIN- q* problem into an instance of the \mathbf{b} -matching problem. The construction of the \mathbf{b} -matching instance will be carried out in two steps. In the first step we build a hypergraph with the properties in Theorem 3.7. In the second step we will, under some conditions,

3.3. The Construction and the Inapproximability Result.

connect the already constructed hypergraphs using a solution of Max-E3-Lin- q to obtain the final instance.

Let $q \in \mathbb{N}_{>1}$ and $\Phi = \{\varphi_1, \dots, \varphi_m\}$ be an instance of the Max-E3-Lin- q over the set of variables $X = \{x_1, \dots, x_n\}$. For every variable $x \in X$ we denote by $c(x)$ the occurrence of the variable x in Φ i.e, the number of equations in which x occurs. Furthermore we denote by $\Phi(x)$ the set of all equations in Φ depending on x i.e., $c(x) = |\Phi(x)|$. According to Theorem 3.4 we may assume that $c(x)$ is constant.

First step: For every variable $x \in X$ we construct a hypergraph, denoted by $H_{b,q}^x$, with the properties as in Theorem 3.7, where the set of vertices is given as follows:

$$V(H_{b,q}^x) = \{(i_x, r) | i_x \in [c(x)], r \in [d]\}$$

Each vertex in $V(H_{b,q}^x)$ corresponds to the appearance-index $i_x \in [c(x)]$, and a number r in $[d]$, where $d = \Theta(q \ln q)$.

The set of hyperedges is defined by:

$$E(H_{b,q}^x) = \{E(i_x, j, s) | (i_x, j) \in [c(x)] \times [q], s \in [b]\}$$

Each hyperedge in $E(H_{b,q}^x)$ is a tuple of three components. The first two of them correspond as in [37] to the number of appearance $i_x \in [c(x)]$ and a number $a \in [q]$ appropriate to the (q, b) -hyperedge coloring of $H_{b,q}^x$, while for the third component we introduce a new index in $[b]$ which we call the *connection-regulator*.

Second step: After having constructed the n hypergraphs according to the variables $x \in X$, we construct the desired hypergraph which will be our instance of the **b**-matching as follows:

The set of vertices is the union of the sets of vertices of the hypergraphs

3. Inapproximability of \mathbf{b} -Matching in l -Uniform Hypergraphs

$H_{b,q}^x$ for $x \in X$ constructed in the first step, so the set of vertices $V(\mathcal{H}_\Phi)$ is given by:

$$V(\mathcal{H}_\Phi) = \bigcup_{x \in X} V(H_{b,q}^x)$$

Let us define the hyperedges of \mathcal{H}_Φ : Let $B \in [q]^{|X|}$ be an assignment. Then we denote by $B(x)$ the value in $[q]$ that will be attributed to the variable $x \in X$ in the assignment B . Let $\varphi \in \Phi$ be an equation on three variables $x, y, z \in X$ and let $B \in [q]^{|X|}$ be an assignment to the variables that satisfies φ .

Finally, the hyperedges that correspond to φ and B will be given as the union of three hyperedges from the hypergraphs $H_{b,q}^x$, $H_{b,q}^y$ and $H_{b,q}^z$ corresponding to the occurrence index of the three variables and having the same *connection-regulator* $s \in [b]$:

$$E(\varphi, B, s) = E(x, i_x, B(x), s) \dot{\cup} E(y, i_y, B(y), s) \dot{\cup} E(z, i_z, B(z), s) \forall s \in [b].$$

In conclusion, the set of hyperedges in \mathcal{H}_Φ is:

$$\mathcal{E}(\mathcal{H}_\Phi) = \{E(\varphi, B, s) | \varphi \in \Phi, B \in [q]^3 \text{ assignment that satisfy } \varphi, s \in [b]\}$$

Example. It is difficult to give an explicit illustration of the constructed instance. In the following example we only try to make the idea of the construction more clear.

Let $b = 2$, $q = 2$ then $d = \Theta(2 \ln 2)$ and for every $x \in X$ with the occurrence number $c(x)$ the above construction gives a hypergraph $H(x)$ with: $V(H(x)) := [c(x)] \times [d]$,

$E(H(x)) := \{e(i_x, 1, 1), e(i_x, 1, 2), e(i_x, 2, 1), e(i_x, 2, 2), \dots\}$ and the suited properties of the structure Theorem.

3.3. The Construction and the Inapproximability Result.

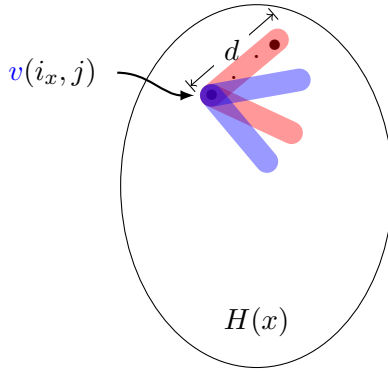


Figure 3.1. In the first phase we construct a hypergraph $H(x)$ which is: d -uniform, 4-regular, $(2, 2)$ -hyperedge colorable and $(2, \frac{1}{4})$ -Partition 2-Matching Concentrated

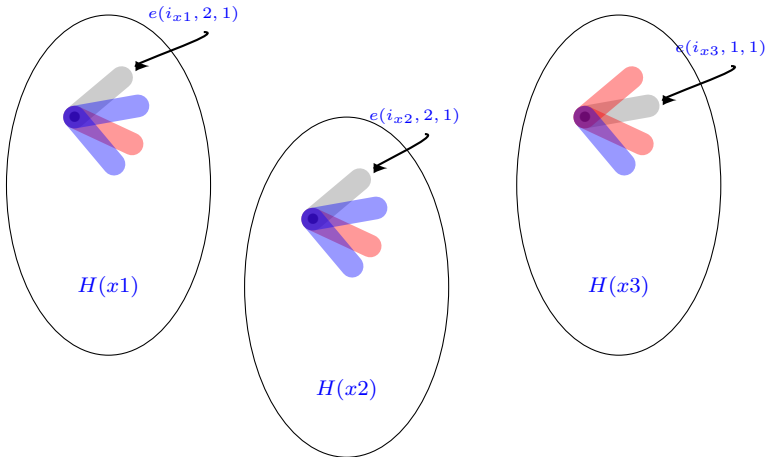


Figure 3.2. In the second phase, we connect the hyperedges of the hypergraphs corresponding to each assignment satisfying φ_1 : For example corresponding to the assignment $(1, 1, 2)$ we get the hyperedge:

$$E(\varphi_1, (1, 1, 2), 1) = e(i_{x1}, 1, 1) \cup e(i_{x2}, 1, 1) \cup e(i_{x3}, 2, 1).$$

3. Inapproximability of \mathbf{b} -Matching in l -Uniform Hypergraphs

This reduction returns the instance \mathcal{H}_Φ of the \mathbf{b} -matching problem in l -uniform hypergraphs where $l = 3d$. According to the Structure Theorem 3.7, for given $b \geq 1$ and $q \geq 2$, since every variable x occurs in Φ a constant number of times, we shall construct all possible hypergraphs with the suitable size in constant time. We enumerate them and verify which of them have the mentioned properties of Theorem 3.7. Therefore, we can construct the instance \mathcal{H}_Φ in polynomial time.

b) Proof of the Non-Approximability Result.

Let $\epsilon > 0$. By Theorem 3.4 we know that the $(\frac{1}{q} + \epsilon, 1 - \epsilon)$ -gap-problem of Max-E3-Lin- q is \mathcal{NP} -hard for the instances considered in the Theorem 3.4.

By the reduction presented in section 3.3 we determine out of the interval $(\frac{1}{q} + \epsilon, 1 - \epsilon)$ an interval (a, b) for that the (a, b) -gap-problem corresponding to the \mathbf{b} -matching problem in uniform hypergraphs is \mathcal{NP} -hard. Then, by applying Corollary 3.2 we get an inapproximability factor for the treated problem.

Next, we will show how to obtain a \mathbf{b} -matching from an assignment of the variables. Then we give the relation between a maximal \mathbf{b} -matching and the maximal number of simultaneously satisfied equations from Φ .

Let us denote by $Sat(\Phi, B)$ the set of all equations in Φ that are satisfied by the assignment B .

The following Lemma proves that each assignment B that satisfies a subsystem in Φ corresponds to a \mathbf{b} -matching in the constructed hypergraph \mathcal{H}_Φ .

3.8 Lemma. *Let $B : X \longrightarrow [q]$ be an assignment. The set $M \subset \mathcal{E}(\mathcal{H}_\Phi)$*

3.3. The Construction and the Inapproximability Result.

defined by:

$$M = \{E(\varphi, B, s) \mid \varphi \in \text{Sat}(\Phi, B), s \in [b]\}$$

is a \mathbf{b} -matching in \mathcal{H}_Φ

Proof. We consider $b + 1$ hyperedges e_1, \dots, e_{b+1} of M and their corresponding equations $\varphi_1, \varphi_2, \dots, \varphi_{b+1}$. Then we can distinguish between the following cases:

- ▷ For every $x \in X$ there are at most b equations $\varphi \in \{\varphi_1, \dots, \varphi_{b+1}\}$ with $\varphi \in \Phi(x)$. Then every vertex in $V(M) \cap H_{b,q}^x$ is contained in at most b hyperedges of M .
- ▷ The equations corresponding to the $b + 1$ hyperedges have a common variable $x \in X$. This means that the hyperedges e_1, \dots, e_{b+1} contain vertices from the hypergraph $H_{b,q}^x$. In $H_{b,q}^x$ each value $B(x)$ is assigned to a color and an index $i \in [q]$. The hyperedges corresponding to $B(x)$ have the same color, furthermore we connect in \mathcal{H}_Φ only these hyperedges with the same connection regulator index s . Hence we know that each vertex in $H_{b,q}^x$ belongs to at most b hyperedges of the same color. So we can assert that if all the $b + 1$ hyperedges have the same color, then at least one of them must have an appearance index different from the others.

The two cases show that no vertex in $V(M)$ is contained in more than b hyperedges of M . This is the definition of a \mathbf{b} -matching. \square

3.9 Lemma. *If there is an assignment B that satisfies $(1 - \epsilon)|\Phi|$ equations, then there exists a \mathbf{b} -matching in \mathcal{H}_Φ with at least $\frac{1-\epsilon}{q^2}|\mathcal{E}(\mathcal{H}_\Phi)|$ hyperedges.*

3. Inapproximability of \mathbf{b} -Matching in l -Uniform Hypergraphs

Proof. Recall that for every equation $\varphi \in \Phi$ there are q^2 satisfying assignments. By the construction of the hypergraph \mathcal{H}_Φ to every assignment satisfying an equation $\varphi \in \Phi$ correspond b hyperedges (because of the connection regulator index $s \in [b]$), So it follows:

$$|\Phi| = \frac{1}{bq^2} |\mathcal{E}(\mathcal{H}_\Phi)|.$$

Let $B \in [q]^{|X|}$ be an assignment that satisfies $(1 - \epsilon)|\Phi|$ equations. Let us consider the subset $M \subset \mathcal{E}$ that consists of hyperedges corresponding to the assignment B

$$M = \{E(\varphi, B, s) \mid \varphi \in \text{Sat}(\Phi, B), s \in [b]\}$$

As proved in Lemma 3.8, M is a \mathbf{b} -matching in \mathcal{H}_Φ . Furthermore we know that to each equation satisfied by B belong b hyperedges in M . So we conclude that:

$$\begin{aligned} |M| &= b(1 - \epsilon)|\Phi| \\ &= \frac{1 - \epsilon}{q^2} |\mathcal{E}(\mathcal{H}_\Phi)| \end{aligned}$$

□

3.10 Lemma. *If every assignment satisfies at most $(\frac{1}{q} + \epsilon)|\Phi|$ equations, then every \mathbf{b} -matching in \mathcal{H}_Φ contains at most $(\frac{3b+1}{q^3} + \epsilon)|\mathcal{E}(\mathcal{H}_\Phi)|$ hyperedges.*

Proof. The main idea in this proof can be described as follows: Let M be a given \mathbf{b} -matching. We define a global-assignment $glob(B)$ for the set of variables X , by giving every variable the value which agrees with the maximal number of hyperedges of M (the values are

3.3. The Construction and the Inapproximability Result.

represented with a color (appropriate to the (q, b) -hyperedge coloring). In this way we partition M into two parts: the first set corresponds to $glob(B)$ and its cardinality can be estimated with help of the construction. The second set consists of the remaining hyperedges in M and its cardinality can be estimated by the (P.b-MC) property. We denote the first part by $glob(M)$ and the second by $rest(M)$. For every $x \in X$ we denote by \mathcal{E}_x the set of hyperedges of \mathcal{H}_Φ corresponding to the equations $\Phi(x)$ and by the set $\mathcal{E}_{x=a}$ a subset of \mathcal{E}_x in which the assignment of x is a .

$$\begin{aligned}\mathcal{E}_x &= \{E(\varphi, B, s) \mid \varphi \in \Phi(x), E(\varphi, B, s) \in \mathcal{E}(\mathcal{H}_\Phi)\} \\ \mathcal{E}_{x=a} &= \{E(\varphi, B, s) \mid E(\varphi, B, s) \in \mathcal{E}_x, B(x) = a\}\end{aligned}$$

Let M be a maximum **b**-matching in \mathcal{H}_Φ . We define the assignment $glob(B)$ according to the matching M as follows: for every $x \in X$, let $glob(B(x))$ be the value $a \in [q]$, so that $|\mathcal{E}_{x=a} \cap M|$ is maximal. Then $glob(M)$ is the subset of M that corresponds to $glob(B)$ and $rest(M)$ is the set of the remaining hyperedges in M :

$$\begin{aligned}glob(M) &= M \cap \{E(\varphi, glob(B), s) \mid \varphi \in \Phi, s \in [b]\} \\ rest(M) &= M \setminus glob(M)\end{aligned}$$

Since the number of equations satisfied by $glob(B)$ is at most $(\frac{1}{q} + \epsilon)|\Phi|$, and for every equation there are bq^2 hyperedges that correspond to all satisfying assignments for this equation, then it holds:

$$|glob(M)| \leq b\left(\frac{1}{q} + \epsilon\right)|\Phi|$$

3. Inapproximability of **b**-Matching in l -Uniform Hypergraphs

$$\stackrel{\text{Lem 3.8}}{=} \left(\frac{1}{q} + \epsilon\right) \frac{|\mathcal{E}(\mathcal{H}_\Phi)|}{q^2}$$

It remains to estimate $|\text{rest}(M)|$. We will use the (P.b-MC) property. Let us consider an equation φ consisting of the three variables x, y, z . We know that the corresponding hypergraphs $H_{b,q}^x$, $H_{b,q}^y$ and $H_{b,q}^z$ have the $(q, \frac{1}{q^2})$ -(P.b-MC) property.

First we observe that $M \cap H_{b,q}^x$ is also a **b**-matching (as every vertex v in $H_{b,q}^x$ is incident with at most b hyperedges). Let $E(\varphi, B, s) \in \text{rest}(M)$ be an hyperedge assigned to the equation φ , depending on x, y and z , $s \in [b]$ and the assignment B . By definition $E(\varphi, B, s)$ is the union of three hyperedges $E(x, i_x, B(x), s)$, $E(y, i_y, B(y), s)$ and $E(z, i_z, B(z), s)$ from $H_{b,q}^x$, $H_{b,q}^y$ and $H_{b,q}^z$:

$$E(\varphi, B, s) = E(x, i_x, B(x), s) \cup E(y, i_y, B(y), s) \cup E(z, i_z, B(z), s)$$

At least one hyperedge of the three is assigned to a value that is not in $\text{glob}(B)$. We assume this is the hyperedge $E(x, i_x, B(x), s)$ from $H_{b,q}^x$. Then the following holds:

$$\sum_{a \neq \text{glob}(B(x))} |\text{rest}(M) \cap \mathcal{E}_{x=a}| \leq \frac{1}{q^2} |\mathcal{E}(H_{b,q}^x)| \quad (3.1)$$

On the other hand, we know that each hyperedge $E(x, i_x, a, s)$ in $H_{b,q}^x$ is used in q hyperedges from \mathcal{H}_Φ , but in a **b**-matching it is not allowed to take more than b hyperedges from the q hyperedges. Then by inequality (3.1), we obtain:

$$\sum_{a \neq \text{glob}(B(x))} |\text{rest}(M) \cap \mathcal{E}_{x=a}| \leq \frac{b}{q^3} |\mathcal{E}_x| \quad (3.2)$$

3.3. The Construction and the Inapproximability Result.

Furthermore this estimate holds for all $x \in X$. This means

$$\begin{aligned}
 |\text{rest}(M)| &\leq \sum_{x \in X, a \neq \text{glob}(B(x))} |\text{rest}(M) \cap \mathcal{E}_{x=a}| \\
 &\leq \frac{b}{q^3} \sum_{x \in X} |\mathcal{E}_x| \\
 &= \frac{3b}{q^3} |\mathcal{E}(\mathcal{H}_\Phi)|
 \end{aligned}$$

and therefore

$$\begin{aligned}
 |M| &= |\text{glob}(M)| + |\text{rest}(M)| \\
 &\leq \left(\frac{1}{q} + \epsilon\right) \frac{|\mathcal{E}(\mathcal{H}_\Phi)|}{q^2} + \frac{3b}{q^3} |\mathcal{E}(\mathcal{H}_\Phi)| \\
 &\leq \left(\frac{3b+1}{q^3} + \epsilon\right) |\mathcal{E}(\mathcal{H}_\Phi)|
 \end{aligned}$$

□

3.11 Theorem. *For every integer $b \geq 1$ and since $l = 3d = \Theta(q \ln q)$ it is \mathcal{NP} -hard to approximate \mathbf{b} -matching on l -uniform hypergraphs to within any ratio of $O\left(\frac{l}{b \ln l}\right)$.*

Proof. By Lemma 3.9 and Lemma 3.10 the $\left(\frac{3b+1}{q^3} + \epsilon, \frac{1}{q^2} - \epsilon\right)$ - \mathbf{b} -matching is \mathcal{NP} -hard. On the other hand we remark that the given interval becomes larger if ϵ is smaller (closer to 0). Let us assume that $\epsilon \leq \frac{1}{q^3}$. By Corollary 3.2 we have

$$\begin{aligned}
 \frac{\frac{1}{q^2} - \epsilon}{\frac{3b+1}{q^3} + \epsilon} &\geq \frac{\frac{1}{q^2} - \frac{1}{2q^2}}{\frac{3b+1}{q^3} + \frac{1}{q^3}} = \frac{\frac{1}{2q^2}}{\frac{3b+2}{q^3}} \\
 &\geq \frac{1}{2q^2} \\
 &\geq_{b \geq 1} \frac{5b}{q^3}
 \end{aligned}$$

3. Inapproximability of \mathbf{b} -Matching in l -Uniform Hypergraphs

$$\geq \frac{1}{10b}q$$

Further we know that $l = 3d = \Theta(q \ln q)$, it means there exist two constants $c_1, c_2 \in \mathbb{R}_{\geq 0}$ such that

$$c_1 q \ln q \leq l \leq c_2 q \ln q$$

The proof of Theorem 3.7 (see next section) provides that $c_1 \geq 8$, than left side of the inequality gives that

$$\begin{aligned} l \geq \frac{c_1}{2}q &\implies \ln l \geq \ln q + \ln\left(\frac{c_1}{2}\right) \\ &\stackrel{c_1 \geq 8}{\implies} \ln l \geq \ln q \end{aligned}$$

Using the right side of the inequality we get:

$$\frac{\frac{1}{q^2} - \epsilon}{\frac{3b+1}{q^3} + \epsilon} \geq \frac{1}{10b}q \geq \frac{l}{10bc_2 \ln q} \geq \frac{l}{10bc_2 \ln l}$$

Therefore we conclude that it is \mathcal{NP} -hard to approximate the \mathbf{b} -matching problem in a l -uniform hypergraph within any ratio of $O(\frac{l}{b \ln l})$. \square

3.4 Proof of the Structure Theorem

For the proof of the structure theorem we follow the strategy as in [37], but we emphasize that essential new ingredients have to be invoked. The proof is divided into two parts. First we randomly generate a hypergraph and in the second part we estimate the probability that the generated hypergraph has the five properties as in Theorem

3.4. Proof of the Structure Theorem

3.7. The main difficulty in the existence proof is to check whether the generated hypergraph possesses the (P.b-MC) property or not.

Proof of Theorem 3.7. Let $t, d \in \mathbb{N}$ (arbitrary). Put $V := [t] \times [d]$ and $V_i := [t] \times \{i\}$ for all $i \in [d]$.

We choose the hyperedges of $H_{b,q}$ randomly, so that the hypergraph is d -uniform and bq -regular.

Let \mathcal{S}_t be the set of all permutations on t elements. We choose uniformly at random for every pair $(j, l) \in [bq] \times [d]$ a permutation $\pi_{j,l}$ from \mathcal{S}_t . The hyperedges are defined as follows:

$$e[i, j] = \{(\pi_{j,1}(i), 1), \dots, (\pi_{j,d}(i), d)\}$$

for all $i \in [t]$, $j \in [bq]$. Finally, the hypergraph is:

$$V := V(H_{b,q}^t), \quad \mathcal{E}(H_{b,q}^t) := \{e[i, j]; (i, j) \in [t] \times [bq]\}$$

It is easy to check, that for every given $q, b \in \mathbb{N}$ there exists a constant $c(q, b)$ so that for every $t \in \mathbb{N}$ with $t \geq c(q, b)$ the generated hypergraph $H_{b,q}$ has a set of hyperedges of cardinality bqt and the three last properties of the structure theorem hold. We claim:

3.12 Claim. For $t \geq bq$ the above construction returns with positive probability a hypergraph with exactly bqt hyperedges.

Proof: Let $t \geq bq$. We consider the subgroup of \mathcal{S}_t generated by $(1\ 2 \dots t)$. Let $\pi \in \langle (1\ 2 \dots t) \rangle$. Then there exists a $l \in \{0, \dots, t-1\}$ with

$$\pi = (1\ 2 \dots t)^l$$

3. Inapproximability of \mathbf{b} -Matching in l -Uniform Hypergraphs

and for all $x \in [t]$ it holds:

$$\pi(x) = (1\ 2 \dots t)^l(x) = x + l \pmod{t}.$$

If $\pi, \sigma \in \langle (1\ 2 \dots t) \rangle$ with $\pi \neq \sigma$, so there exists $l, m \in \{0, \dots, t-1\}$ with $l \neq m$, $\pi = (1\ 2 \dots t)^l$ and $\sigma = (1\ 2 \dots t)^m$. For all $x \in [t]$ it holds:

$$\pi(x) = x + l \pmod{t}$$

and

$$\sigma(x) = x + m \pmod{t}.$$

Because of that $l \neq m$ and $l, m \in \{0, \dots, t-1\}$ the following holds:

$$x + l \not\equiv x + m \pmod{t},$$

this means

$$\pi(x) \neq \sigma(x) \text{ for all } x \in [t]. \quad (*)$$

We consider the following event: For each pair $(j, k) \in [bt] \times [d]$ let

$$\pi_{j,k} = \begin{cases} (1\ 2 \dots t)^{j-1} & , \text{ if } k = 1 \\ id & , \text{ otherwise} \end{cases}$$

be the at random choosing permutation from \mathcal{S}_t . The probability that this event occurs is strict greater than 0. Now we prove that, if this event occurs then it holds $|E| = bqt$.

It is clear that $|\mathcal{E}| \leq bqt$ (definition of \mathcal{E}). Let $(i, j), (\tilde{i}, \tilde{j}) \in [t] \times [bt]$ with $(i, j) \neq (\tilde{i}, \tilde{j})$. If we prove that $e[i, j] \neq e[\tilde{i}, \tilde{j}]$, than $|\mathcal{E}| = bqt$ holds, too.

3.4. Proof of the Structure Theorem

case 1: $i \neq \tilde{i}$

It holds:

$$(i, 2) = (id(i), 2) = (\pi_{j,2}(i), 2) \in e[i, j]$$

and

$$(\tilde{i}, 2) = (id(\tilde{i}), 2) = (\pi_{\tilde{j},2}(\tilde{i}), 2) \in e[\tilde{i}, \tilde{j}].$$

As $i \neq \tilde{i}$, it follows $(i, 2) \neq (\tilde{i}, 2)$. By the definition of the hyperedges, we obtain that

$$(i, 2) \notin e[\tilde{i}, \tilde{j}]$$

and

$$(\tilde{i}, 2) \notin e[i, j].$$

Therewith it follows $e[i, j] \neq e[\tilde{i}, \tilde{j}]$.

case 2: $i = \tilde{i}$

It follows $j \neq \tilde{j}$ and it holds:

$$(\pi_{j,1}(i), 1) \in e[i, j]$$

and

$$(\pi_{\tilde{j},1}(i), 1) \in e[\tilde{i}, \tilde{j}].$$

By the fact that $j \neq \tilde{j}$, for $j, \tilde{j} \in [bq]$ and $bq \leq t$ it follows:

$$\pi_{j,1} = (1 \ 2 \ \dots \ t)^{j-1} \neq (1 \ 2 \ \dots \ t)^{\tilde{j}-1} = \pi_{\tilde{j},1}.$$

By (*) follows therewith

$$\pi_{j,1}(i) \neq \pi_{\tilde{j},1}(i),$$

3. Inapproximability of **b**-Matching in l -Uniform Hypergraphs

this gives

$$(\pi_{j,1}(i), 1) \neq (\pi_{\tilde{j},1}(i), 1)$$

by the definition of the hyperedges and according to $e[i, j] \neq e[\tilde{i}, \tilde{j}]$ follows that $|\mathcal{E}| = bqt$. This concludes the proof of Claim 3.12

Now we are able to prove the remaining properties of the hypergraph $H_{b,q}$.

1. Uniformity and regularity of $H_{b,q}$:

Each hyperedge of $\mathcal{E}(H_{b,q})$ has exactly d vertices. This implies the d -uniformity of $H_{b,q}$.

2. Regularity of $H_{b,q}$:

Since $\mathcal{E}(H_{b,q}) = bqt$, define the sets

$\mathcal{E}_i := \{e(j, i); j \in [t]\}$ for all $i \in [bq]$. Thus $|\mathcal{E}_1| = \dots = |\mathcal{E}_{bq}| = t$ and every hyperedge of a set \mathcal{E}_i covers exactly one vertex $v \in V_i$. That leads to the bq -regularity of the hypergraph.

3. (q, b) -hyperedge colorability of $H_{b,q}$:

For the (q, b) -hyperedge colorability of $H_{b,q}$ we may color the hyperedges as follows:

Since we have bq disjoint sets and every hyperedge covers exactly one vertex, than for every $i \in [bq]$ we pick from every set \mathcal{E}_i one hyperedge and include it in a new set $\tilde{\mathcal{E}}_r$ for $r \in [q]$. It is obvious that the set $\tilde{\mathcal{E}}_r$ is of cardinality bt . Finally we color all hyperedges in a set $\tilde{\mathcal{E}}_r$ with the color r .

In this way we obtain a partition of $\mathcal{E}(H_{b,q})$ into q subsets:

$$\mathcal{E}(H_{b,q}) = \tilde{\mathcal{E}}_1 \dot{\cup} \dots \dot{\cup} \tilde{\mathcal{E}}_q \text{ with } |\tilde{\mathcal{E}}_k| = bt \text{ for } k \in [q].$$

3.4. Proof of the Structure Theorem

Also we obtain a function (coloration) $f : \mathcal{E} \rightarrow [q]$ such that every vertex is contained in exactly b hyperedges of the same color which yields the (q, b) -coloring property of the hyperedges.

4. The (P.b-MC) property:

We now prove the (P.b-MC) property with the probabilistic methods. We claim

3.13 Claim. With high probability the hypergraph $H_{b,q}$ possesses the (P.b-MC) property.

Proof. Before we start with the proof of the above claim, we need to state some useful facts.

Define for all $k \in [q]$ the set \mathcal{M}_k as follows:

$$\mathcal{M}_k = \left\{ M \subset \mathcal{E}; |M \cap \mathcal{E}_k| \leq \frac{bt}{q}, |M \setminus \mathcal{E}_k| = \frac{bt}{q}, \forall i \in [q] |M \cap \mathcal{E}_k| \geq |M \cap \mathcal{E}_i| \right\}$$

3.14 Proposition. *If the hypergraph $H_{b,q}$ does not have the (P.b-MC) property, then there exists a $k \in [q]$ and a set $M \in \mathcal{M}_k$ that is a \mathbf{b} -matching.*

Proof. If $H_{b,q}$ does not have the (P.b-MC) property, then there exists a \mathbf{b} -matching $M' \subset \mathcal{E}$ that is not concentrated in one of the $\tilde{\mathcal{E}}_j$. Formally: $|M' \setminus \mathcal{E}_j| > \frac{1}{q^2} |\mathcal{E}| = \frac{bt}{q}$ for all $j \in [q]$. Let $k \in [q]$, such that $|M' \cap \mathcal{E}_k|$ is maximal. We know that a subset of a \mathbf{b} -matching is a \mathbf{b} -matching, so this allows us to delete enough hyperedges until there remain exactly $\frac{bt}{q}$ in $M' \setminus \mathcal{E}_j$ for all $i \neq k$ and $M' \cap \mathcal{E}_k$ contains at most $\frac{bt}{q}$. We denote this new set by M

3. Inapproximability of **b**-Matching in l -Uniform Hypergraphs

and we claim that for all $j \in [q]$ the inequality

$$|M \cap \mathcal{E}_k| \geq |M \cap \mathcal{E}_j|$$

holds.

Case 1: $|M' \cap \mathcal{E}_k| \leq \frac{bt}{q}$

Then we will delete no hyperedges from $M' \cap \mathcal{E}_k$ and it follows that:

$$\begin{aligned} |M \cap \mathcal{E}_k| &= |M' \cap \mathcal{E}_k| \\ &\geq |M' \cap \mathcal{E}_j| \\ &\geq |M \cap \mathcal{E}_j| \quad \text{for all } j \in [q] \end{aligned}$$

Case 2: $|M' \cap \mathcal{E}_k| > \frac{bt}{q}$

Then we have

$$|M \cap \mathcal{E}_k| = \frac{bt}{q}$$

and it follows that:

$$\begin{aligned} |M \setminus \mathcal{E}_k| &= |M| - |M \cap \mathcal{E}_k| \\ &= \left(\sum_{j=1}^q |M \cap \mathcal{E}_j| \right) - |M \cap \mathcal{E}_k| \quad \text{since } \mathcal{E} = \mathcal{E}_1 \dot{\cup} \dots \dot{\cup} \mathcal{E}_q \\ &= \sum_{[q] \setminus \{k\}} |M \cap \mathcal{E}_j| \end{aligned}$$

therefore we obtain

$$\sum_{[q] \setminus \{k\}} |M \cap \mathcal{E}_j| = |M \setminus \mathcal{E}_k| = \frac{bt}{q}$$

3.4. Proof of the Structure Theorem

and finally it holds:

$$|M \cap \mathcal{E}_j| \leq \frac{bt}{q} = |M \cap \mathcal{E}_k| \quad \text{for all } j \in [q].$$

This implies that M is in \mathcal{M}_k .

This concludes the proof of the Proposition 3.14

3.15 Claim. For every $k \in [q]$ let \mathcal{M}_k be defined as above. It holds:

$$|\mathcal{M}_k| \leq \frac{bt}{q} q^{\frac{3bt}{q}} e^{\frac{2bt}{q}} \quad (3.3)$$

Proof. The set \mathcal{M}_k is bounded above by the number of possibilities to choose at most $\frac{bt}{q}$ hyperedges from $\tilde{\mathcal{E}}_k$ and the remainder $\frac{bt}{q}$ hyperedges from the remaining sets. This means:

$$|\mathcal{M}_k| \leq \binom{b(q-1)t}{\frac{bt}{q}} \sum_{j=1}^{\frac{bt}{q}} \binom{bt}{j}$$

on the other hand we know that

$$\binom{n}{s} \leq \left(\frac{en}{s}\right)^s.$$

holds and that the function

$$\begin{aligned} f : \{0, \dots, \frac{n}{2}\} &\longrightarrow \mathbb{Q} \\ s &\longmapsto \binom{n}{s} \end{aligned}$$

is monoton increasing, hence $|\mathcal{M}_k| \leq \frac{bt}{q} (q)^{\frac{3bt}{q}} e^{\frac{2bt}{q}}$.

3. Inapproximability of **b**-Matching in l -Uniform Hypergraphs

This concludes the proof of Claim 3.15.

Now we continue the proof of Theorem 3.7. Let \mathbb{P} be the probability that $H_{b,q}$ does not have the (P.b-MC) property. In the following we consider only the set \mathcal{M}_1 . Then due to the symmetry of \mathcal{M}_k according to k we have:

$$\begin{aligned} \mathbb{P} &\leq \Pr(\exists k \in [q] : \exists M \in \mathcal{M}_k, M \text{ is a b-matching}) \\ &\leq q \sum_{M \in \mathcal{M}_1} \Pr(M \text{ is a b-matching}) \\ &\leq q |\mathcal{M}_1| \Pr(\tilde{M}) \end{aligned} \tag{3.4}$$

where $\tilde{M} \in \mathcal{M}_1$ with $\Pr(\tilde{M}) = \max_{M \in \mathcal{M}_1} \Pr(M \text{ is a b-matching})$. According to Claim 3.15 we know how large is \mathcal{M}_1 . It remains to estimate the term $\Pr(\tilde{M})$.

For all $i \in [d]$ we denote by $\mathcal{A}_i(\tilde{M})$ the event that for every $v \in \tilde{M} \cap V_i$ (the set of vertices of \tilde{M} restricted to V_i) there are at most b incident hyperedges in \tilde{M} . Due to the independence of the choice of the permutations in the construction of the hypergraph $H_{b,q}$ the events $\mathcal{A}_i(\tilde{M})$ are independent so

$$\Pr(\tilde{M}) = \prod_{i=1}^d \Pr(\mathcal{A}_i(\tilde{M})) \tag{3.5}$$

Since the events $\mathcal{A}_i(\tilde{M})$ are identically distributed, it suffices to estimate the probability of one of the $\mathcal{A}_i(\tilde{M})$, for example $\mathcal{A}_1(\tilde{M})$. Therewith we obtain

$$\prod_{i=1}^d \Pr(\mathcal{A}_i(\tilde{M})) = \left(\Pr(\mathcal{A}_1(\tilde{M})) \right)^d.$$

It is not trivial for us to estimate the probability of the event

3.4. Proof of the Structure Theorem

$\mathcal{A}_1(\tilde{M})$. Namely if we consider only the case of $b = 2$, then for every vertex $v \in V_1$ we get two events that depend on each other:

- (a) v belongs to two hyperedges of M of different colors, or
- (b) v belongs to two hyperedges of M of the same color.

To solve this problem we modify slightly the resulting instance in the following way:

Algorithm 2: Instance Modification

1. initialise $\hat{M} := \tilde{M}$, $\hat{V} := V_1$, $j := 0$.
 2. for every $v \in \hat{M} \cap V_1$ and $i \in [q]$.
Make a copy v' and pick arbitrary an hyperedge
 $E_i \in \hat{M} \cap \Gamma(v)$
 3. set $e' := (e_i \setminus \{v\})$, $\hat{M} := (\hat{M} \setminus \{e_i\}) \cup \{e'\}$, $\hat{V} := V_1 \cup \{v\}$
and $j := j + 1$.
 4. if $j < b - 1$ and $|\hat{M} \cap \Gamma(v)| > 1$ then goto 2.
 5. return \hat{M} and \hat{V} .
-

Description: Let us consider the set of vertices V_1 . For every vertex $v \in \tilde{M} \cap V_1$ we consider the set of hyperedges that contain v i.e., $\{e_i \in \tilde{M} \cap \Gamma(v) : v \in e_i\}$. We make a copy v' of v , delete the hyperedge e_i and add a new hyperedge $e'_i = (e_i \setminus \{v\}) \cup \{v'\}$, which gets the same color as e_i (i.e., the same index $i \in [q]$).

For every vertex $v \in V_1$ the procedure will be $b - 1$ times repeated if the number of the incidence hyperedges to v in the new set of

3. Inapproximability of \mathbf{b} -Matching in l -Uniform Hypergraphs

hyperedges \hat{M} not earlier becomes equal to one.

For both sets provided by the above modification, we have $|\hat{M}| = |\tilde{M}|$ and $|\hat{V}| \leq b|V_1|$, so we could claim that, if \hat{M} restricted to \hat{V} is a 1-matching, then \tilde{M} is a \mathbf{b} -matching restricted to V_1 and vice versa.

If $\mathcal{A}(\hat{M})$ is the event, that \hat{M} restricted to \hat{V} is a 1-matching, then

$$\Pr(\mathcal{A}_1(\tilde{M})) \leq \Pr(\mathcal{A}(\hat{M}))$$

Now let us consider the set of vertices from \hat{V} that belong to the hyperedges from $\hat{M} \cap \mathcal{E}_i$. Formally:

$$\hat{M}_i = \left(\bigcup (\hat{M} \cap \mathcal{E}_i) \right) \cap \hat{V} \text{ for all } i \in [q]$$

For every $i \in [q]$, let now A_i be the event that the sets of hyperedges $\{\hat{M}_j | j \leq i\}$ are disjoint i.e., for $i \in [q]$, A_i is the event that the sets $\hat{M}_1, \dots, \hat{M}_i$ are pairwise disjoint. Then the following holds:

$$\Pr(\mathcal{A}(\hat{M})) = \Pr(\bigcap_{i=2}^q A_i) = \prod_{i=2}^q \Pr(A_i | A_{i-1}) \quad (3.6)$$

Moreover for every $i \in \{2, \dots, q\}$ the probability that the event $A_i | A_{i-1}$ occurs, is equal to the probability of choosing $|\hat{M}_i|$ different vertices from \hat{V} at random without the set $\cup_{l=1}^{i-1} \hat{M}_l$.

On the other hand we know that the probability to choose $|\hat{M}_i|$ different vertices from \hat{V} without the set $\cup_{l=1}^{i-1} \hat{M}_l$ is smaller than the probability to choose $|\hat{M}_i|$ vertices with repetition. Furthermore the occurrence of A_i implies that the sets \hat{M}_l are disjoint for all

3.4. Proof of the Structure Theorem

$l < i$, and this implies that $|\cup_{l=1}^{i-1} \hat{M}_l| = \sum_{l=1}^{i-1} |\hat{M}_l|$. So we have:

$$\begin{aligned}
 \Pr(A_i | A_{i-1}) &\leq \left(1 - \frac{\sum_{l < i} |\hat{M}_l|}{|\hat{V}|}\right)^{|\hat{M}_i|} \\
 &\leq \left(1 - \frac{\sum_{l < i} |\hat{M}_l|}{bt}\right)^{|\hat{M}_i|} \\
 &\leq \exp\left(-\frac{1}{bt} |\hat{M}_i| \sum_{l < i} |\hat{M}_l|\right) \tag{3.7}
 \end{aligned}$$

and we have

$$\begin{aligned}
 \Pr(\tilde{M}) &\leq \exp\left(-\frac{d}{bt} \sum_{i=2}^q |\hat{M}_i| \sum_{j=1}^{i-1} |\hat{M}_j|\right) \\
 &= \exp\left(-\frac{d}{bt} \sum_{j < i} |\hat{M}_i| |\hat{M}_j|\right) \tag{3.8}
 \end{aligned}$$

We know that under the condition $\tilde{M} \in \mathcal{M}_1$ the following holds:

$$|\hat{M}_1| = |\tilde{M}_1| \geq \max_{i=2}^q |\tilde{M}_i|$$

and

$$\sum_{i=2}^q |\hat{M}_i| = \sum_{i=2}^q |\tilde{M}_i| = |\tilde{M} \setminus \tilde{\mathcal{E}}_k| = \frac{bt}{q}.$$

We invoke the following claim:

3.16 Claim. Let $q \in \mathbb{N}$, $x_1, \dots, x_q \geq 0$, $x_1 \geq \max_{i=2}^q x_i$ and

3. Inapproximability of **b**-Matching in l -Uniform Hypergraphs

$T := \sum_{i=2}^q x_i$. Then it holds that:

$$\sum_{0 \leq i < j \leq q} x_i x_j \geq \frac{1}{4} T^2.$$

Proof. We distinguish the two following cases:

▷ If $x_1 \geq \frac{T}{2}$, we get

$$\begin{aligned} \sum_{1 \leq i < j \leq q} x_i x_j &= \sum_{1 < j \leq q} x_1 x_j + \sum_{1 < i < j \leq q} x_i x_j \\ &\geq \sum_{1 < j \leq q} x_1 x_j = x_1 \sum_{1 < j \leq q} x_j \\ &\geq \frac{T}{2} T \\ &\geq \frac{T^2}{2}. \end{aligned}$$

▷ If $x_1 \leq \frac{T}{2}$ then it holds:

$$\begin{aligned} \sum_{1 \leq i < j \leq q} x_i x_j &\stackrel{(x_1, \dots, x_q \geq 0)}{\geq} \sum_{2 \leq i < j \leq q} x_i x_j \\ &\stackrel{\text{binom. formula}}{=} \frac{1}{2} \left(\sum_{i=2}^q x_i \right)^2 - \frac{1}{2} \sum_{i=2}^q x_i^2 \\ &\geq \frac{T^2}{2} - \sum_{i=2}^q x_i^2, \end{aligned}$$

Further we consider the following function:

$$f: \mathbb{R}^{q-1} \rightarrow \mathbb{R}, (y_2, \dots, y_{q-1}) = \sum_{i=2}^q y_i^2.$$

3.4. Proof of the Structure Theorem

and the set

$$S := \{y \in \mathbb{R}^{q-1} : \sum_{i=2}^q y_i = T \text{ and } \max_{i=2}^q y_i \leq \frac{T}{2}\}$$

By the method of Lagrange multipliers, we get that the function f restricted to the set S achieves its maximum in

$$y_2 = y_3 = \frac{T}{2} \quad \text{and} \quad y_4 = \dots, y_q = 0.$$

Therefore we obtain:

$$\sum_{i=1}^{q-1} x_i^2 \leq \frac{1}{4}T^2,$$

hence we have,

$$\sum_{1 \leq i < j \leq q} x_i x_j \geq \frac{1}{4}T^2.$$

This concludes the Claim 3.16.

By Claim 3.16 we have:

$$\sum_{j < i} |\hat{M}_i| |\hat{M}_j| \geq \frac{(bt)^2}{4(q)^2}. \quad (3.9)$$

Applying (3.9) to inequality (3.8) we obtain:

$$\Pr(\tilde{M}) \leq \exp\left(-\frac{d}{bt} \frac{(bt)^2}{4q^2}\right) \quad (3.10)$$

3. Inapproximability of **b**-Matching in l -Uniform Hypergraphs

By the inequalities (3.3), (3.4) and (3.10) it follows:

$$\begin{aligned} \mathcal{P} &\leq bt \cdot (q)^{\frac{3bt}{q}} \exp\left(\frac{2bt}{q} - \frac{d}{bt} \frac{(bt)^2}{4(q)^2}\right) \\ &= bt \cdot \exp\left(\frac{3bt}{q} \ln q + \frac{2bt}{q} - \frac{dbt}{4q^2}\right) \end{aligned}$$

For every d , that satisfies the inequality

$$bt \cdot \exp\left(\frac{3bt}{q} \ln q + \frac{2bt}{q} - \frac{dbt}{4q^2}\right) < 1$$

the hypergraph has property (P.b-MC) with positive probability.

For example for $d > 32q \ln q$, we have

$$\begin{aligned} d &> 32q \ln q = 4q \ln q + 16q \ln q + 12q \ln q \\ &\geq 4q^2 \frac{\ln q}{q} + 16q \ln q + 12q \ln q \\ &\stackrel{q \geq 2, 2 \ln q \geq 1}{\geq} 4q^2 \frac{\ln q}{q} + 8q + 12q \ln q, \end{aligned}$$

since the function $x \rightarrow \frac{\ln x}{x}$ is monotone decreasing for all $x \in \mathbb{R}_{\geq e}$, we get

$$\begin{aligned} d &> 4q^2 \frac{\ln bt}{bt} + 8q + 12q \ln q \\ &= \frac{4q^2}{bt} \left(\ln bt + 2 \frac{bt}{q} + 3 \frac{bt}{q} \ln q \right), \end{aligned}$$

and it follows

$$\frac{dbt}{4q^2} > \ln bt + 2 \frac{bt}{q} + 3 \frac{bt}{q} \ln q,$$

3.5. Summary and Future Work

this yields

$$0 > \ln bt + 2\frac{bt}{q} + 3\frac{bt}{q} \ln q - \frac{dbt}{4q^2},$$

and finally we get

$$bt \cdot \exp\left(\frac{3bt}{q} \ln q + \frac{2bt}{q} - \frac{dbt}{4q^2}\right) < 1$$

□

3.5 Summary and Future Work

The main contribution of this chapter is the proof that it is \mathcal{NP} -hard to approximate the \mathbf{b} -matching problem in a l -uniform hypergraph within any ratio smaller than $O(\frac{l}{b \ln l})$ (Theorem 3.7).

This shows that the approximation depends on b and the non-approximability ratio tends towards a constant if b tends to $\frac{l}{\ln l}$. In other words if b is getting larger, much better approximation ratios might be possible compared to the case of $b = 1$. It is known that an approximation ratio of $O(l)$ for the \mathbf{b} -matching problem can be achieved [56]. This is still far from our inapproximability lower bound, but we now know the interval in which improvements of the $O(l)$ -approximation can happen. Our lower bound indicates that better approximation algorithms should invoke b in the approximation ratio. The structure of the proof consists of two parts:

- (a) first, we prove the existence of a uniform hypergraph with some partition properties of \mathbf{b} -matching hyperedges,
- (b) secondly, taking hypergraphs from (a), a final instance for \mathbf{b} -matching from a solution to the Max-E3-Lin- q problem must be

3. Inapproximability of \mathbf{b} -Matching in l -Uniform Hypergraphs

build.

Both steps require some new ideas to cope with $b \geq 2$.

In Section 3.3 we describe a reduction from an instance of Max-E3-Lin- q to a \mathbf{b} -matching problem instance. The basis for the reduction is: to each variable x in the instance of Max-E3-Lin- q we assign a hypergraph with some suitable properties. The existence of such hypergraphs is proved by the probabilistic method. Afterwards depending on each equation in Φ with a given assignment, some hyperedges of the three hypergraphs are connected to achieve the desired instance. In the case $b = 1$ one hyperedge is obtained for every equation φ and a satisfying assignment to φ and for every assignment B to Φ the set of hyperedges corresponding to B is a 1-matching. If we would construct the hyperedges of the hypergraph in the case $b \geq 2$ in the same manner as Hazan, Safra and Schwartz, then for a given assignment to Φ the set corresponding to this assignment would not be a \mathbf{b} -matching, because there are b^3 incident hyperedges to every vertex. To avoid this infeasibility we introduce a new hyperedge parameter $s \in [b]$ called *connection-regulator*: only the hyperedges with the same *connection-regulator* will be connected. In this way the hypergraph is sparse enough.

For the probabilistic proof of the existence of hypergraphs with the required properties we follow the strategy as in [37], but with some essential modifications to cope with the \mathbf{b} -matching structure. After generating a hypergraph at random in the same way as in [37] we wish to color the hypergraph in such a way that every vertex is contained in at most b hyperedges of the same color.

The 1-matching condition and the property that no vertex is covered by two hyperedges of the same color implies that for a subset M

3.5. Summary and Future Work

of the hyperedges with the same color are disjoint. Thus, M is a 1-matching if and only if each pair of differently colored hyperedges of M do not intersect (according to the strong hyperedge colorability, see [37]). Such a nice characterization is not possible for $b \geq 2$. The deeper reason is that in contrast to $b = 1$ events have to be considered which are highly *dependent* on each other. For example in the case of $b = 2$ we have two dependent events i. e.,: M is a 2-matching if and only if every vertex is contained in at most

- ▷ two hyperedges of M of different colors or
- ▷ in two hyperedges of M of the same color.

To estimate the probability of the above events is difficult. We resolve this problem by dispersing the \mathbf{b} -matching in a deterministic way so that the \mathbf{b} -matching case is reduced to 1-matching. Now the probabilistic method works.

There remain interesting open questions concerning the \mathbf{b} -matching problem in l -uniform hypergraphs. According to our inapproximability result we see two main directions for research:

- ▷ The range from $b \in [\frac{l}{\ln l}, \ln n]$ is almost unexplored.
- ▷ The gap between the recently best approximation ratio of $O(l)$ due to Krysta [56] and our inapproximation ratio of $O(\frac{l}{b \ln l})$ is still large. This motivates us to look for a tighter approximation ratio which would involve b .

The \mathbf{b} -Matching Problem in l -Uniform Hypergraphs is MAX-SNP-Hard

In this chapter we denote by $\mathbf{x} = (x_1, \dots, x_n)$ a tuple in \mathbb{N}^n where for every $i \in [n]$, $x_i = x$ (a positive integer). Our goal is to prove that the restriction of the \mathbf{b} -matching problem to a certain class of hypergraphs has no polynomial-time approximation scheme PTAS, unless $\mathcal{P} = \mathcal{NP}$. We utilize the concept of L -reduction (see [60]) instead of directly using a PTAS-reduction, because it is often easier to show that a reduction is an L -reduction than a PTAS-reduction. The main part of this chapter¹ deals with the \mathbf{b} -matching problem on the subclass of l -uniform hypergraphs with vertex degree at most a constant Δ .

However, concerning its algorithmic complexity, the problem has still not been investigated extensively, which motivated our study. The only known result of the problem related to the class MAX-SNP is due to Kann [47], who proved that there is no approximation scheme for the case $b = 1$, unless $\mathcal{P} = \mathcal{NP}$.

The chapter is organized as follows. Section 4.1 contains some definitions and preliminaries. In Section 4.2 we present our results and give an L -reduction from an instance of the MAX-3-SAT to an instance

¹This chapter is mainly based on the paper [22]

4. The \mathbf{b} -Matching Problem in l -Uniform Hypergraphs is MAX-SNP-Hard

of the \mathbf{b} -matching problem. In Section 4.3 we analyze the presented reduction. In Section 4.4 we give a reduction of the constructed instance to a l -uniform hypergraph. In Section 4.5 we give further example of L-reductions in the esperance to clarify better the terminology of the L-reductions and the MAX-SNP class. In Section 4.6 we give a summary of the chapter and cite some future works.

4.1 Tools and Definitions

For the convenience of the reader, we recall some definitions from previous chapters.

Formulation of the Treated Problems

\mathbf{b} -Matching Problem: For given $\mathbf{b} = (b_1, \dots, b_n) \in \mathbb{N}^n$ we call a set $M_{\mathbf{b}} \subseteq \mathcal{E}$ a **\mathbf{b} -matching** if no $i \in V$ vertex is contained in more than b_i edges of $M_{\mathbf{b}}$. **\mathbf{b} -MATCHING Problem** is the problem of finding a \mathbf{b} -matching with maximum cardinality. We denote the \mathbf{b} -matching problem in hypergraphs with bounded degree Δ by (\mathbf{b}, Δ) -matching.

Remark. In this chapter we are interested of the case where, for every vertex $i \in V$, $b_i = b$ (b is a positiv integer)

l -Dimensional \mathbf{b} -Matching Problem: This problem is a variant of \mathbf{b} -matching in l -uniform hypergraphs, where the vertices of the input hypergraph are a union of l disjoint sets, $V = V_1 \dot{\cup} V_2 \dots \dot{\cup} V_l$, and each hyperedge contains exactly one vertex from each set such that we have $E \subseteq V_1 \times V_2 \times \dots \times V_l$. We will denote this problem in hypergraphs with bounded degree Δ by l -dimensional (\mathbf{b}, Δ) -matching problem

Set \mathbf{d} -Multicover Problem: For given $\mathbf{d} = (d_1, \dots, d_n) \in \mathbb{N}^n$ we call $S_{\mathbf{d}} \subseteq \mathcal{E}$ a set multicover if no vertex $i \in V$ is contained in less than d_i

4.1. Tools and Definitions

hyperedges of S_d . Set multicover problem is the problem of finding a set multicover with minimum cardinality.

Remark. In this chapter we are interested in the case where for every vertex $i \in V$, $d_i = d$ (d is a positive integer)

The following reduction was introduced by Papadimitriou and Yannakakis [60].

4.1 Definition. Let $P = (X, (S_x)_{x \in X}, w)$ and $P' = (X', (S'_x)_{x \in X}, w')$ be two optimization problems with non-negative weights. An L -reduction from P to P' is a pair of functions f and g , both computable in polynomial-time, and two constants $\alpha, \beta \in \mathbb{R}^+$ such that it holds for any instance x of P :

1. $f(x)$ is an instance of P' with $\text{Opt}(f(x)) \leq \alpha \cdot \text{Opt}(x)$,
2. For any feasible solution y' of $f(x)$, $g(x, y')$ is a feasible solution of x such that $|w(g(x, y')) - \text{Opt}(x)| \leq \beta \cdot |w'(y') - \text{Opt}(f(x))|$.

We say that P is L -reducible to P' if there is an L -reduction from P to P' .

Note that “ L ” stands for “linear”. The L -reduction can be composed as follows:

4.2 Proposition. *Let P_1, P_2, P_3 be optimization problems with non-negative weights. If $(f_1, g_1, \alpha_1, \beta_1)$ is an L -reduction from P_1 to P_2 and $(f_2, g_2, \alpha_2, \beta_2)$ is an L -reduction from P_2 to P_3 , then their composition $(f_3, g_3, \alpha_1\alpha_2, \beta_1\beta_2)$ is an L -reduction from P_1 to P_3 , where $f_3(x) = f_2(f_1(x))$ and $g_3(x, y_3) = g_1(x, g_2(f_1(x), y_3))$.*

To introduce the MAX-SNP-hardness of an optimization problem we have to define the maximum 3-satisfiability problem:

4. The \mathbf{b} -Matching Problem in l -Uniform Hypergraphs is MAX-SNP-Hard

4.3 Definition. Consider a set $X = \{x_1, \dots, x_n\}$ of variables and a family $\mathcal{C} = \{c_1, \dots, c_m\}$ of different clauses, each involving exactly three literals (a variable or a negated variable) over X .

- a) Maximum 3-satisfiability (in short MAX-3-SAT) is the problem of finding an assignment A of X so that the number of clauses in \mathcal{C} satisfied by A is maximum.
- b) Maximum bounded 3-satisfiability- Δ (in short MAX-3-SAT- Δ) is a variant of MAX 3-SAT, where the number of occurrences of each variable is bounded by a constant $\Delta \in \mathbb{N}$.

The following fundamental theorem is due to Papadimitriou and Yannakakis [60]

4.4 Definition. An optimization problem P with non-negative weights is called MAX-SNP-hard if MAX-3-SAT is L -reducible to P .

4.5 Corollary. *No PTAS for any MAX-SNP-hard problem exists, unless $\mathcal{P} = \mathcal{NP}$.*

4.6 Theorem. *Maximum bounded 3-satisfiability- Δ problem is MAX-SNP-hard.*

The following theorem due to Ray-Chaudhuri [69] is very important for the reduction from the minimum set multicover problem to the b -matching problem.

4.7 Theorem (Ray-Chaudhuri, 1960). *Consider a hypergraph \mathcal{H} with maximum vertex degree Δ not necessary bounded and $\mathbf{b}, \mathbf{d} \in \mathbb{N}^n$ such that $b_i + d_i = \deg i$, where $\deg(i)$ is the degree of the vertex i . A subset of hyperedges $M_{\mathbf{b}}$ is a \mathbf{b} -matching in H if and only if the subset $S_{\mathbf{d}} := \mathcal{E} \setminus M_{\mathbf{b}}$ is a set \mathbf{d} -multicover in H . Furthermore $M_{\mathbf{b}}$ is maximal if and only if $S_{\mathbf{d}}$ is minimal.*

4.2 Main Result

In the following we will list the main results of this chapter. The next theorem will be shown by an L -reduction from MAX-3-SAT- Δ to 3-dimensional (\mathbf{b}, Δ) -matching.

4.8 Theorem. *For every $\mathbf{b} \in \mathbb{N}^n$ with $b \leq \frac{\Delta}{3}$, the 3-dimensional (\mathbf{b}, Δ) -matching problem is MAX-SNP-hard.*

Proof. To prove Theorem 4.8, we will describe an L -reduction (f, g, α, β) from a MAX-3-SAT- Δ problem to 3-dimensional (\mathbf{b}, Δ) -matching problem. This is an extension of the reduction used to prove that 3-dimensional $(\mathbf{1}, \Delta)$ -matching problem is MAX-SNP-hard in [47]. The reduction consists of two steps. We consider an instance I of MAX-3-SAT- Δ with a set of variables $X = \{x_1, \dots, x_n\}$ and a set of clauses $\mathcal{C} = \{c_1, \dots, c_m\}$.

First step: We make $b - 1$ copies of \mathcal{C} in a way to obtain b identical sets of clauses $\mathcal{C}^1, \dots, \mathcal{C}^b$. For every set of clauses \mathcal{C}^j , $j \in [b]$, this part of the construction remains the same as in [47] and can be described as follows:

We consider a set of clauses $\mathcal{C}^j = \{c_1^j, \dots, c_m^j\}$. We assume that every variable $x_i \in X$ appears $c(x_i)$ times in \mathcal{C}^j (either as x_i or as \bar{x}_i). Moreover, let $K = 2^{\lfloor \log_2 \frac{3}{2} b \Delta + 1 \rfloor}$ be the largest power of 2 such that $K \leq \frac{3}{2} b \Delta + 1$. For every variable x_i we construct K identical rings of triples (in a triangle form) that we denote by $R(x_i, k, j)$, $k \in [K]$. Each ring contains $2c(x_i)$ triples. The free vertices in the ring triples (the apices of the triangles) are denoted by $x(\rho, \lambda, j)$ and $\bar{x}(\rho, \lambda, j)$, where $(\rho, \lambda, j) \in [c(x_i)] \times [K] \times [b]$ (see Figure 4.1).

The K rings $R(x_i, 1, j), \dots, R(x_i, K, j)$ are connected by tree triples

4. The **b**-Matching Problem in l -Uniform Hypergraphs is MAX-SNP-Hard

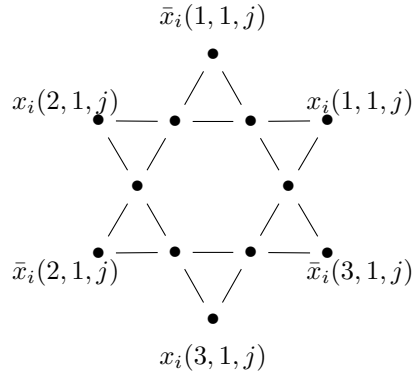


Figure 4.1. The ring $R(x_i, 1, j)$, the first of K rings for the set of clauses \mathcal{C}^j for a variable x_i with $c(x_i) = 3$ occurrences.

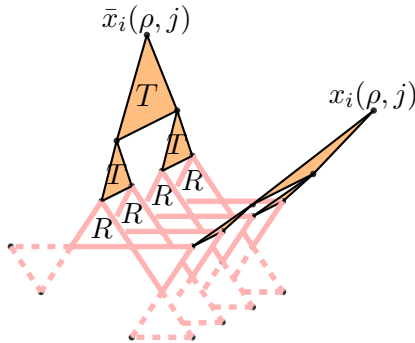


Figure 4.2. An example of binary trees of triples with $c(x_i) = 3$ and $K = 4$.

4.2. Main Result

in $2c(x_i)$ binary trees, denoted by $T_{x_i(\rho,j)}$ and $T_{\bar{x}_i(\rho,j)}$, ($\rho \in [c(x_i)]$) in such a way that $x_i(\rho, 1, j), x_i(\rho, 2, j), \dots, x_i(\rho, K, j)$ are leaves in the tree $T_{x_i(\rho,j)}$ and $\bar{x}_i(\rho, 1, j), \bar{x}_i(\rho, 2, j), \dots, \bar{x}_i(\rho, K, j)$ are leaves in the tree $T_{\bar{x}_i(\rho,j)}$ (see Figure 4.2).

The root of $T_{x_i(\rho,j)}$ is $x_i(\rho, j)$ and the root of $T_{\bar{x}_i(\rho,j)}$ is $\bar{x}_i(\rho, j)$. We denote by $\hat{T}_{x_i(\rho,j)}$ and $\hat{T}_{\bar{x}_i(\rho,j)}$ the binary tree consisting of $T_{x_i(\rho,j)}$ and $T_{\bar{x}_i(\rho,j)}$, respectively, and the K ring triples with the nodes $x_i(\rho, 1, j), \dots, x_i(\rho, K, j)$ and $\bar{x}_i(\rho, 1, j), \dots, \bar{x}_i(\rho, K, j)$.

Finally, the clause triples connect some of the roots. For each clause c_l^j , $l \in [m]$, we introduce two new elements $s_1(l, j)$ and $s_2(l, j)$. If the variable x_i occurs in this clause and this is its ρ -th occurrence in \mathcal{C}^j , then we connect $s_1(l, j)$, $s_2(l, j)$ with the root element $x_i(\rho, j)$ or $\bar{x}_i(\rho, j)$, depending on whether the occurrence is x_i or \bar{x}_i . (see Figure 4.3).

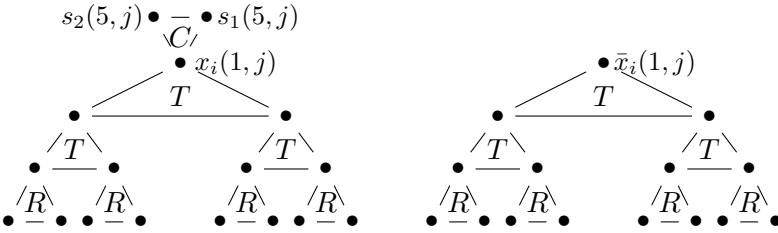


Figure 4.3. An example of binary trees for x_i and the adjacent clause triple and ring triples, where the first occurrence of x_i in \mathcal{C}^j is in the 5-th clause. The triples are marked with R , T and C for ring, tree and clause, respectively.

Second step: The construction in the first step holds for every set of clauses \mathcal{C}^j for all $j \in [b]$.

4. The **b**-Matching Problem in l -Uniform Hypergraphs is MAX-SNP-Hard

Let us now consider two constructed binary trees $\hat{T}_{x_i(\rho,j)}$ and $\hat{T}_{x_i(\rho,j')}$ corresponding to the same variable $x_i \in X$ and the same occurrence number $\rho \in [c(x_i)]$, but belonging to two different sets of clauses \mathcal{C}^j and $\mathcal{C}^{j'}$, analogously for trees $\hat{T}_{\bar{x}_i(\rho,j)}$ and $\hat{T}_{\bar{x}_i(\rho,j')}$. Furthermore let L be the number of levels of $\hat{T}_{x_i(\rho,j)}$ and $\hat{T}_{x_i(\rho,j')}$. We consider a triple (triangle) T in $\hat{T}_{x_i(\rho,j)}$. Then the apex of T belongs to levels r for an $r \in \{0, \dots, L-1\}$ and the two vertices from the base belong to level $r+1$. In the tree $\hat{T}_{x_i(\rho,j')}$ we have exactly one triangle T' between levels r and $r+1$ corresponding to T . We connect the apex of T with two vertices of T' in level $r+1$ and the apex of T' with two vertices of T in level $r+1$ to obtain two new triangles. The same will be done for the clause triples $(s_1(l,j), s_2(l,j), x_i(\rho,j))$ and $(s_1(l,j'), s_2(l,j'), x_i(\rho,j'))$ in order to obtain two new clause triples $(s_1(l,j), s_2(l,j), x_i(\rho,j'))$ and $(s_1(l,j'), s_2(l,j'), x_i(\rho,j))$. This process is applied to every pair of symmetric triangles for every pair of symmetric trees $\hat{T}_{x_i(\rho,j)}$ and $\hat{T}_{x_i(\rho,j')}$. Finally, we obtain two new binary trees $\hat{T}_{x_i(\rho,j \rightarrow j')}$ and $\hat{T}_{x_i(\rho,j' \rightarrow j)}$ between each pair of binary trees such that the set of their vertices alternates between the sets of vertices of the trees $\hat{T}_{x_i(\rho,j)}$ and $\hat{T}_{x_i(\rho,j')}$ (or $\hat{T}_{\bar{x}_i(\rho,j)}$ and $\hat{T}_{\bar{x}_i(\rho,j')}$) (see Figure 4.4).

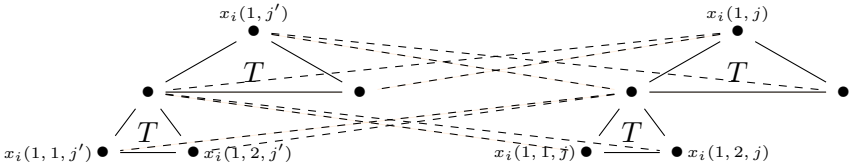


Figure 4.4. An example of connecting the elements between two symmetric trees, namely $\hat{T}_{x_i(\rho,j)}$ and $\hat{T}_{x_i(\rho,j')}$ in two different sets of clauses \mathcal{C}^j and $\mathcal{C}^{j'}$.

The constructed instance contains $b^2 \sum_{x \in X} c(x)$ clause triples,

4.2. Main Result

$2Kb^2 \sum_{x \in X} c(x)$ ring triples and $2b^2(K-1) \sum_{x \in X} c(x)$ tree triples. To obtain the final instance, namely a 3-dimensional hypergraph, we label the vertices of the constructed instance. This will be done first separately for every clause set \mathcal{C}^j , $j \in [b]$. In order to define $E^j(\mathcal{H}) \subseteq Z^j \times U^j \times Y^j$ for ring, tree and clause triples corresponding to a \mathcal{C}^j we have to label elements of Z^j , U^j or Y^j . It suffices to label the ring, tree and clause triples obtained in the first step since the rest of triples will be automatically labelled.

All trees $\hat{T}_{x_i(\rho, j)}$ and $\hat{T}_{\bar{x}_i(\rho, j)}$ are labelled identically as follows. Start with the root and label it with Z^j and label the elements in every tree triple Z^j , U^j and Y^j anti-clockwise. The ring triples are labelled anti-clockwise in x_i -trees and clockwise in \bar{x}_i -trees. The elements $s_1(l, j)$ and $s_2(l, j)$ are labelled with U^j and Y^j , respectively (see Figure 4.5).

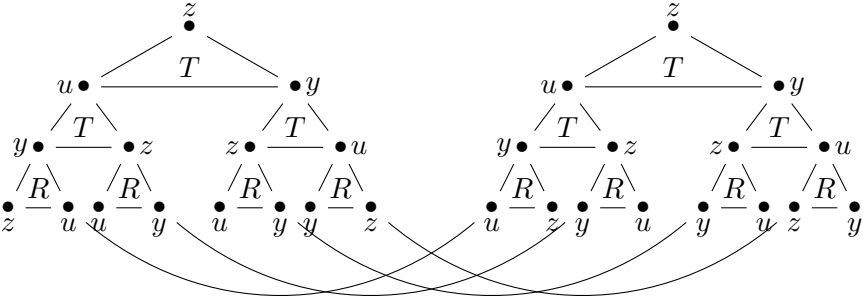


Figure 4.5. An example of element labelling in a tree with two levels. Dots which represent identical elements are connected with arcs.

For the final instance we set $X := \bigcup_{j \in [b]} X^j$, $Y := \bigcup_{j \in [b]} Y^j$ and $Z := \bigcup_{j \in [b]} Z^j$ and $E(\mathcal{H}) \subseteq Z \times U \times Y$ contains all triples obtained

4. The **b**-Matching Problem in l -Uniform Hypergraphs is MAX-SNP-Hard

at the end of second step.

Explanations to the constructed instance.

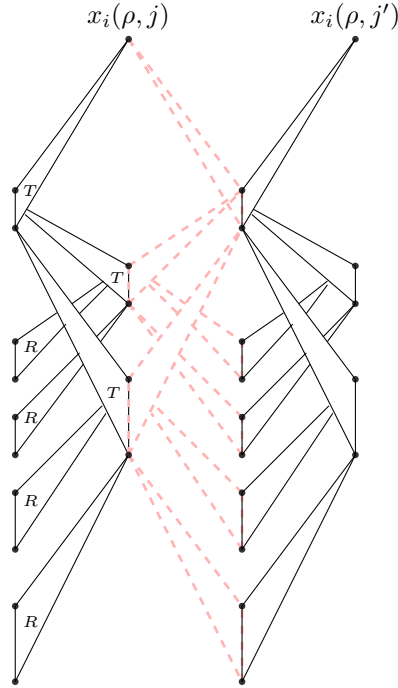


Figure 4.6. An example of binary trees and rings of triples with $K = 4$. The tree on the left side of the figure is the tree $\hat{T}_{x_i(\rho, j)}$ corresponding to the variable x_i and the set of clauses C^j . The tree on the right side is the tree $\hat{T}_{x_i(\rho, j')}$ corresponding to the same variable and the set of clauses $C^{j'}$. The tree in the middle is the tree $T_{x_i(\rho, j \rightarrow j')}$ that alternates between $\hat{T}_{x_i(\rho, j)}$ and $\hat{T}_{x_i(\rho, j')}$ starting from the root $x_i(\rho, j)$.

Before proceeding we give a summary of the somewhat complicated construction. Every vertex $x_i(\rho, j)$ appropriate to its occurrence in a set of clauses C^j represents a root of b binary trees $\hat{T}_{x_i(\rho, j)}$ and

4.3. Analysis of the Reduction

$\hat{T}_{x_i(\rho, j \rightarrow j')}$ for $j' \in [b] \setminus \{j\}$, where $\hat{T}_{x_i(\rho, j)}$ is the tree defined above and $\hat{T}_{x_i(\rho, j \rightarrow j')}$ are the $b - 1$ binary trees that leave the root $x_i(\rho, j)$ and alternate between $\hat{T}_{x_i(\rho, j)}$ and $\hat{T}_{x_i(\rho, j')}$ in a zick-zack fashion from the level r in $\hat{T}_{x_i(\rho, j)}$ to the level $r + 1$ in $\hat{T}_{x_i(\rho, j')}$ and return to the level $r + 2$ in $\hat{T}_{x_i(\rho, j)}$ (see Figure 4.6). We note that for every $j \in [b]$ the sets of binary trees

$(\hat{T}_{x_i(\rho, j)} \cup \bigcup_{j' \in [b] \setminus \{j\}} \hat{T}_{x_i(\rho, j \rightarrow j')})$ and $(\hat{T}_{x_i(\rho, j')} \cup \bigcup_{j \in [b] \setminus \{j'\}} \hat{T}_{x_i(\rho, j' \rightarrow j)})$ are symmetric. Therefore, it suffices to focus our analysis on an arbitrary set of clauses \mathcal{C}^j . To depict the following analysis, we define for every $x_i \in X$, $\rho \in [c(x_i)]$ and $k \in [K]$ the sets $\mathcal{T}_{x_i, j}$, $\hat{\mathcal{T}}_{x_i, j}$ and $\mathcal{R}_{x_i, j}$ as follows:

$$\begin{aligned} \mathcal{T}_{x_i, j} &= \bigcup_{\rho \in [c(x_i)]} \left(T_{x_i(\rho, j)} \cup \bigcup_{j' \in [b] \setminus \{j\}} T_{x_i(\rho, j \rightarrow j')} \right) \\ \hat{\mathcal{T}}_{x_i, j} &= \bigcup_{\rho \in [c(x_i)]} \left(\hat{T}_{x_i(\rho, j)} \cup \bigcup_{j' \in [b] \setminus \{j\}} \hat{T}_{x_i(\rho, j \rightarrow j')} \right) \\ \mathcal{R}_{x_i, j} &= \hat{\mathcal{T}}_{x_i, j} \setminus \mathcal{T}_{x_i, j} \end{aligned}$$

By $\mathcal{R}_{x_i, j}(k)$ we denote the set $\mathcal{R}_{x_i, j}$ restricted to the k -th set of ring triples.

4.3 Analysis of the Reduction

The presented reduction is of polynomial-time since the returned instance \mathcal{H} has a set of hyperedges of cardinality at most $9b^2m + 6b^3\Delta^2n$:

$$|E(\mathcal{H})| = \overbrace{b^2 \sum_{x \in X} c(x)}^{\text{clause triples}} + \overbrace{2Kb^2 \sum_{x \in X} c(x)}^{\text{ring triples}} + \overbrace{2b^2(K-1) \sum_{x \in X} c(x)}^{\text{tree triples}},$$

4. The \mathbf{b} -Matching Problem in l -Uniform Hypergraphs is MAX-SNP-Hard

furthermore we have

$$\sum_{i=1}^n c(x_i) \leq 3m. \quad (4.1)$$

Then

$$\begin{aligned} |E(\mathcal{H})| &\stackrel{\text{Eq (4.1)}}{\leq} 3b^2m + b^2 \sum_{i=1}^n 2c(x_i)(K + K - 1) \\ &\stackrel{K \leq \frac{3}{2}b\Delta + 1}{\leq} 3b^2m + b^2 \sum_{i=1}^n 2c(x_i)(3b\Delta + 1) \\ &\leq 3b^2m + b^2 \left(6b\Delta \sum_{i=1}^n c(x_i) + 2 \sum_{i=1}^n c(x_i) \right) \\ &\stackrel{c(x) \leq \Delta}{\leq} 3b^2m + b^2 \left(6b\Delta^2 n + 2 \sum_{i=1}^n c(x_i) \right) \\ &\stackrel{\text{Eq (4.1)}}{\leq} 3b^2m + b^2 6b\Delta^2 n + 6b^2m = b^2(9m + 6b\Delta^2 n). \end{aligned}$$

We conclude that f can be computed in time polynomial in m and n . Moreover, expect for half of the root elements that occur b times, every element in a ring or tree triples occur exactly $2b$ times in \mathcal{H} . The elements are $s_1(l, j)$ and $s_2(l, j)$ in the clause triples occur at most $3b$ times each, because a clause contains at most three literals and is connected with b root elements in the b sets of clauses. Thus, $f(I)$ is an instance of the Max 3-dimensional (\mathbf{b}, Δ) -matching problem for $\Delta \geq 3b$.

To compute the parameters α and β from the definition of L -reduction, we introduce the so-called standard \mathbf{b} -matching:

4.9 Definition. A \mathbf{b} -matching M in $f(I)$ is called **standard \mathbf{b} -matching** if its ring and tree triples are matched in an even distance. We denote it by $stand(M)$. Moreover, for every $i \in [n]$ an optimal standard

4.3. Analysis of the Reduction

b-matching contains either all ring triples corresponding to x_i or all corresponding to \bar{x}_i (see Figure 4.7).

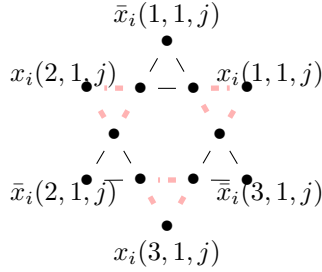


Figure 4.7. An example of matched ring triples (with dashed edges) in an optimal standard 1-matching for $c(x_i) = 3$.

The following assertions are useful to understand the subsequent analysis.

Assertion 1. In an optimal **b**-matching of the structure of ring triples, all ring triples belonging to the same variable are matched in the same way. Moreover, any maximum **b**-matching of the whole problem contains a maximum **b**-matching of this structure.

Proof: Let M be a given 3-dimensional **b**-matching in $f(I)$. For every variable x_i we consider the sets $M \cap \mathcal{R}_{x_i,j}(1)$ and $M \cap \mathcal{R}_{x_i,j}(2)$. Suppose that the first set has cardinality t_1 and the second set has cardinality t_2 and w.l.o.g., $t_1 \geq t_2$. Since the two sets of ring triples have an empty cut ($\mathcal{R}_{x_i,j}(1) \cap \mathcal{R}_{x_i,j}(2) = \emptyset$), we are able to construct on $\mathcal{R}_{x_i,j}(1) \cup \mathcal{R}_{x_i,j}(2)$ a new **b**-matching that is larger than the original one by matching the triples in $\mathcal{R}_{x_i,j}(2)$ like $\mathcal{R}_{x_i,j}(1)$ — this depends on how the neighbour sets of tree triples $\mathcal{R}_{\bar{x}_i,j}(1)$ respectively $\mathcal{R}_{\bar{x}_i,j}(2)$ on left and right of the set $\mathcal{R}_{x_i,j}(1)$ respectively $\mathcal{R}_{x_i,j}(1)$ — . This increases its cardinality to t_1 and also increases the **b**-matching. This will

4. The \mathbf{b} -Matching Problem in l -Uniform Hypergraphs is MAX-SNP-Hard

be done for every adjacent set of ring triples in the matching. for every set of tree triples in the lowest level, we pick some triples in the matching while the \mathbf{b} -matching constraints are not violate

Once this is done, we consider now two adjacent pairs of sets of rings $(M \cap \mathcal{R}_{x_i,j}(1), M \cap \mathcal{R}_{x_i,j}(2))$ and $(M \cap \mathcal{R}_{x_i,j}(3), M \cap \mathcal{R}_{x_i,j}(4))$ and its connecting tree triples situated in the lowest level of $\mathcal{T}_{x_i,j}$. We suppose as above that the cardinality of the first pair and its connecting tree triples is t_1 and of the second pair and its connecting tree triples is t_2 with $t_1 \geq t_2$. In this way along the K rings, we can gain a \mathbf{b} -matching from the sets of rings that is larger than the original one.

We can continue on $\hat{\mathcal{T}}_{x_i,j}$ with the same procedure as described above. In each iteration we go one level up and continue in $\log_2 K - 1$ steps in the same manner.

Now we are done with constructing a new \mathbf{b} -matching M' from the given \mathbf{b} -matching M on the structure of ring triples and tree triples $\hat{\mathcal{T}}_{x_i,j}$, and we want to extend the constructed new \mathbf{b} -matching over the entire instance. Therefore, we have to carefully include some triples from the clause triples. We suppose that we cannot include all the p clause triples belonging to the \mathbf{b} -matching M in the new \mathbf{b} -matching M' . This case occurs if in M' there are some roots $x_i(\rho', j)$ for $\rho' \in [c(x_i)]$ with \mathbf{b} incident tree triples.

The easiest way to include all the p clause triples is by sacrificing some other tree or ring triples. W.l.o.g., sacrifice δ ring triples to include the p clauses triples in the \mathbf{b} -matching.

Moreover, the cardinality of the \mathbf{b} -matching on the structure of ring and tree triples is the cardinality of the matched ring triples plus the cardinality of the tree triples in the tree connected to the matched

4.3. Analysis of the Reduction

ring plus the cardinality of the matched tree triples in trees connected to non-matched ring triples. In the case of an odd number of levels this cardinality is smaller than

$$\begin{aligned} & ((bc(x_i) - \delta) \cdot K + ((bc(x_i) - \delta) \cdot \frac{K-2}{3} + (bc(x_i) + \delta) \cdot \frac{2K-1}{3}) \\ = & bc(x_i)(2K-1) - \delta \cdot \frac{2K-1}{3}. \end{aligned}$$

In the case of an even number of levels this cardinality is less than

$$\begin{aligned} & ((bc(x_i) - \delta) \cdot K + ((bc(x_i) - \delta) \cdot \frac{K-1}{3} + (bc(x_i) + \delta) \cdot \frac{2K-2}{3}) \\ = & bc(x_i)(2K-1) - \delta \cdot \frac{2K+1}{3}. \end{aligned}$$

This estimations show how many triples compared with a maximum **b**-matching we can lose if we sacrifice δ ring triples. If we don't sacrifice any ring triples, i.e., $\delta = 0$, we obtain a maximum **b**-matching of cardinality $bc(x_i) \cdot (2K-1)$.

It is obvious that every ring triple included in the **b**-matching is more valuable than the inclusion of some clause triples. Therefore we conclude that any maximum **b**-matching of the whole instance $f(I)$ contains a maximum **b**-matching in $f(I)$ without clause triples.

To extend a given maximum **b**-matching of $\hat{\mathcal{T}}_{x_i,j}$ over the set of clause triples in order to gain a large **b**-matching over the whole instance, we have to include as many of the clause triples as possible. For every clause in \mathcal{C}^j at most b triples can be included. Depending on how the substructure $\hat{\mathcal{T}}_{x_i(\rho,j)}$ is matched, we include $b - r(\rho)$ with $r(\rho) \leq b$ clause triples if $r(\rho)$ triples of $\hat{\mathcal{T}}_{x_i(\rho,j)}$ are included that contain the root $x_i(\rho, j)$. \square

4. The **b**-Matching Problem in l -Uniform Hypergraphs is MAX-SNP-Hard

Assertion 2. To every maximum **b**-matching M on the structure of ring and tree triples we can construct a corresponding standard **b**-matching $Stand(M)$ in polynomial time.

Proof: A standard matching of cardinality $bc(x_i)(2K - 1)$ on $\hat{\mathcal{T}}_{x_i,j}$ is easy to construct, depending of the given **b**-matching and can be described as follows:

Let M be a **b**-matching in $f(I)$ without clause triples. For every $x_i \in X$ we consider the restriction of M on the ring triples. If $|M \cap \mathcal{R}_{x_i,j}| \geq |M \cap \mathcal{R}_{\bar{x}_i,j}|$, then we include for every vertex $x_i(\rho, \lambda, j)$ for $(\rho, \lambda, j) \in [c(x_i)] \times [K] \times [b]$, b incident triples from the sets of ring triples $\mathcal{R}_{x_i,j}$ and delete all ring triples in $M \cap \mathcal{R}_{\bar{x}_i,j}$. Otherwise the same is applied to \bar{x}_i . This will be done for every $j \in [b]$ separately. Thereafter, let n_{x_i} be the number of rings within which all matching triples belong to x_i and $n_{\bar{x}_i}$ the number of rings within which all matching triples belong to \bar{x}_i . If $n_{x_i} \geq n_{\bar{x}_i}$ we match all rings that belong to \bar{x}_i similar to those rings that belong to x_i . Otherwise, we match all rings that belong to x_i similar to those rings that belong to \bar{x}_i . W.l.o.g assume that we deal with the first case than for every $j \in [b]$ we continue as follows.

Suppose that the set of tree triples $\mathcal{T}_{x_i,j}$ (i.e., binary tree corresponding to x_i without ring and clause triples), and $\mathcal{T}_{\bar{x}_i,j}$, respectively, has L levels. Then we include from $\mathcal{T}_{x_i,j}$ all triples in level $L - 1$ and from $\mathcal{T}_{\bar{x}_i,j}$ all triples in level L . This procedure is applied along both sets of trees by going two levels up in each step. As K and the occurrence for every x_i are constants ($K, c(x_i) = O(1)$), the construction of $stand(M)$ can be done in deterministic polynomial-time. As a property of the standard b -matching is that both ways of

4.3. Analysis of the Reduction

including triples (whether x_i or \bar{x}_i) correspond to the truth values of x_i , we are done. \square

4.10 Lemma. *The transformation*

$$f : \text{MAX-3-SAT-}\Delta \longrightarrow \text{MAX-3-dimensional } (\mathbf{b}, \Delta)\text{-matching,}$$

is an L-reduction, where $\alpha = b^2(18b\Delta + 7)$ and $\beta = 1$.

Proof: Let I be an instance of MAX-3-SAT- Δ . Then it is always possible to satisfy at least $\frac{m}{2}$ clauses of I . On the other hand we can construct for every optimal \mathbf{b} -matching in $f(I)$ a corresponding standard \mathbf{b} -matching that is optimal. Thus the following holds

$$\begin{aligned} \text{Opt}(f(I)) &\leq b^2 \sum_{i=1}^n (c(x_i)(2K - 1)) + b^2 \text{Opt}(I) \\ &= b^2(2K - 1) \sum_{i=1}^n c(x_i) + b^2 \text{Opt}(I) \\ &\stackrel{K \leq \frac{3}{2}b\Delta + 1}{\leq} b^2 \left(\left(2 \left(\frac{3}{2}b\Delta + 1 \right) - 1 \right) \sum_{i=1}^n c(x_i) + \text{Opt}(I) \right) \\ &\stackrel{\text{Eq (4.1)}}{\leq} b^2 (3m (3b\Delta + 1) + \text{Opt}(I)) \\ &\leq b^2 \left(6 \frac{m}{2} (3b\Delta + 1) + \text{Opt}(I) \right) \\ &\stackrel{\text{Opt}(I) \geq \frac{m}{2}}{\leq} b^2 (6 (3b\Delta + 1) \text{Opt}(I) + \text{Opt}(I)) \\ &\leq b^2(18b\Delta + 7) \text{Opt}(I). \end{aligned}$$

Therefore, $\alpha = b^2(18b\Delta + 7)$ satisfies the first constraint of an L -reduction.

Furthermore, for every \mathbf{b} -matching M of the cardinality c_2 we can

4. The \mathbf{b} -Matching Problem in l -Uniform Hypergraphs is MAX-SNP-Hard

construct in polynomial-time a solution of I with c_1 satisfied clauses and $\text{Opt}(f(I)) - c_2 \geq \beta^{-1}(\text{Opt}(I) - c_1)$, where $\beta^{-1} = 1$. As explained above, if a given \mathbf{b} -matching M restricted on $\hat{\mathcal{T}}_{x_i,j}$ is not optimal, we can make it optimal on this substructure. We presume that the \mathbf{b} -matching $M|_{\hat{\mathcal{T}}_{x_i,j}}$ is optimal. We construct a standard optimal \mathbf{b} -matching $\text{Stand}(M|_{\hat{\mathcal{T}}_{x_i,j}})$ over $\hat{\mathcal{T}}_{x_i,j}$ corresponding to M . Thus it follows:

$$\text{Opt}(f(I)) - |M|_{\hat{\mathcal{T}}_{x_i,j}} = \text{Opt}(f(I)) - |\text{Stand}(M|_{\hat{\mathcal{T}}_{x_i,j}})|.$$

We set the variables of I , as the \mathbf{b} -matching $\text{Stand}(M|_{\hat{\mathcal{T}}_{x_i,j}})$ indicates. By looking at the ring triples in the \mathbf{b} -matching, we obtain an approximate solution to I that satisfies c_1 clauses and $\text{Opt}(f(I)) - c_2 \geq (\text{Opt}(I) - c_1)$. \square

4.4 The Extension to l -uniform hypergraphs

From the construction and Lemma 4.3, Theorem 4.8 follows.

4.11 Theorem. *There exists an L -Reduction from the 3-dimensional (\mathbf{b}, Δ) -matching problem to the (\mathbf{b}, Δ) -matching problem in l -uniform hypergraphs with $\alpha = \beta = 1$.*

Proof: It is easy to transform a 3-dimensional (\mathbf{b}, Δ) -matching instance to an instance of \mathbf{b} -matching in a 3-uniform hypergraph, where the set of vertices is the union of the three sets of the partition, and the set of hyperedges is still the same. This transformation is an L -reduction with $\alpha = 1$ and $\beta = 1$. If we compose the above two

4.5. Further L -reduction

transformations, we obtain an L -reduction f' from the bounded MAX-3-SAT- Δ to the bounded (\mathbf{b}, Δ) -matching restricted to 3-uniform hypergraphs with the same α and β as given in the first reduction. So we conclude that \mathbf{b} -matching restricted to 3-uniform hypergraphs is also MAX-SNP-hard.

For $l > 3$, by extending all hyperedges of the constructed 3-uniform hypergraph to l elements and by introducing extra dummy elements, we obtain also an L -reduction with $\alpha = 1$ and $\beta = 1$. With the same argument, by composing this transformation and f' , we obtain a new L -reduction with α and β like for f . \square

The following corollary can easily conclude from Theorems 4.8 and 4.11:

4.12 Corollary. *The (\mathbf{b}, Δ) -matching problem in l -uniform hypergraphs is MAX-SNP-hard.*

Proof. By Theorem 4.8, Theorem 4.11 and Proposition 4.2 we get an L -reduction from MAX-3-SAT- Δ to the (\mathbf{b}, Δ) -matching problem in l -uniform hypergraphs with $\alpha = b^2(18\Delta + 7)$ and $\beta = 1$. Therefore by Theorem 4.6 we conclude that the (\mathbf{b}, Δ) -matching problem in l -uniform hypergraphs for $b \leq \frac{\Delta}{3}$ is MAX-SNP-hard. \square

4.5 Further L -reduction

Let $\Delta = (\Delta_1, \dots, \Delta_n) \in \mathbb{N}^n$ where for every $i \in [n]$, $\Delta_i = \Delta$ a positive integer. In this section we show the existence of an L -reduction from the set multicover problem to the $(\Delta - \mathbf{d})$ -matching problem for regular hypergraphs with vertex degree Δ not necessary bounded and of an L -reduction from the \mathbf{b} -matching problem to the set $(\Delta - \mathbf{b})$ -multicover problem in Δ -regular hypergraphs with

4. The **b**-Matching Problem in l -Uniform Hypergraphs is MAX-SNP-Hard

maximum size of all hyperedges at most a constant l . Hence, if we prove that one of the problems does not have a PTAS, then this holds also for the second one.

4.13 Theorem. *For a regular hypergraph with vertex degree Δ and $d \in \mathbb{N}$ with $d \geq \frac{\Delta}{2}$, there is an L -reduction from the set **d**-multicover problem to the $(\Delta - \mathbf{d})$ -matching problem.*

Proof: Let $\mathcal{H} = (V, \mathcal{E})$ be a regular hypergraph of order n with degree Δ , and let $d \in \mathbb{N}$ with $d \geq \frac{\Delta}{2}$. Consider the transformation from the set multicover problem with $f(\mathcal{H}) := \mathcal{H}$ and $g(\mathcal{H}, X) := \mathcal{E}(\mathcal{H}) \setminus X$ for all $X \subseteq \mathcal{E}(\mathcal{H})$. Set $b := \Delta - d$, and denote by $\nu_{\mathbf{b}}$ and $\rho_{\mathbf{d}}$ the maximum cardinality of a **b**-matching in \mathcal{H} and the minimum cardinality of a set multicover in \mathcal{H} , respectively. Furthermore, let $M_{\mathbf{b}}$ be a **b**-matching of cardinality $|M_{\mathbf{b}}|$ and let $S_{\mathbf{d}}$ be a set **d**-multicover of cardinality $|S_{\mathbf{d}}|$. By Theorem 4.7, we have

$$\nu_{\mathbf{b}} = |\mathcal{E}| - \rho_{\mathbf{d}} = |\mathcal{E} \setminus S_{\mathbf{d}}| + |S_{\mathbf{d}}| - \rho_{\mathbf{d}} = |M_{\mathbf{b}}| + |S_{\mathbf{d}}| - \rho_{\mathbf{d}}.$$

and it follows $\nu_{\mathbf{b}} - |M_{\mathbf{b}}| = |S_{\mathbf{d}}| - \rho_{\mathbf{d}}$.

On the other hand, it is clear that $\nu_{\mathbf{b}} \leq \rho_{\mathbf{d}}$ since $d \geq \frac{\Delta}{2}$. This shows that the given transformation is an L -reduction from the set **d**-multicover problem in \mathcal{H} to the **b**-matching problem in \mathcal{H} with $\beta_1 = \alpha_1 = 1$. \square

4.14 Theorem. *Let \mathcal{H} be a Δ -regular hypergraph with maximum size of all hyperedges at most a constant l and $b \in \mathbb{N}$ with $b \leq \frac{\Delta}{2}$. Then there is an L -reduction from the **b**-matching problem to the set $(\Delta - \mathbf{b})$ -multicover problem.*

Proof. It remains to find an α such that $\rho_{\mathbf{d}} \leq \alpha \nu_{(\Delta - \mathbf{d})}$, since the

Algorithm 3: \mathbf{b} -Matching Greedy

Input : A hypergraph $\mathcal{H} = (V, \mathcal{E})$ and $\mathbf{b} = (b, \dots, b) \in \mathbb{N}^n$ **Output**: A \mathbf{b} -matching set $M_{\mathbf{b}}$

1. Initialize $M^0 := \emptyset$, $\mathcal{E}_0 := \mathcal{E} = \{E_1, \dots, E_m\}$, $i := 0$.
 2. Choose $E_{i+1} \in \mathcal{E}_i$ and set $M^{i+1} := M^i \cup \{E_{i+1}\}$.
 3. Set $\mathcal{E}_{i+1} := \mathcal{E}_i \setminus (C(M^{i+1}) \cup \{E_{i+1}\})$.
 4. If $|\mathcal{E}_{i+1}| > 0$, then increment i and goto 2.
 5. Return $M_{\mathbf{b}} := M^{i+1}$ and the number of iterations $t := i + 1$.
-

way to find β is the same as in Theorem 4.13 holds.

Let us consider the Algorithm 4, which returns a \mathbf{b} -matching set. Before proceeding, we define some notations that we need for our analysis.

For each $Z \subseteq E$ define the set of conflicting edges to Z by

$$C(Z) := \{E \in \mathcal{E}; \exists v \in E : \deg(v, Z \cup \{E\}) > b\} .$$

Thus, for a \mathbf{b} -matching $M_{\mathbf{b}}$ we have that $C(M_{\mathbf{b}})$ is the set of edges that cannot be added to $M_{\mathbf{b}}$ without violating its \mathbf{b} -matching property.

Claim 3: Algorithm \mathbf{b} -Matching Greedy constructs a \mathbf{b} -matching $M_{\mathbf{b}}$ with $|M_{\mathbf{b}}| \geq \frac{1}{t(\Delta-b+1)+1} |\mathcal{E}|$.

Proof of Claim 3: It is clear by construction that $M_{\mathbf{b}}$ is a \mathbf{b} -matching (of cardinality t). In each iteration $j = 1, \dots, t$, the set $D_{E_j} := (C(M^j) \cup \{E_j\}) \cap \mathcal{E}_{j-1}$ is removed from \mathcal{E}_{j-1} to form the new set of remaining edges \mathcal{E}_j . The sets D_{E_1}, \dots, D_{E_t} form a partition of \mathcal{E} . Let

4. The \mathbf{b} -Matching Problem in l -Uniform Hypergraphs is MAX-SNP-Hard

$j \in [t]$ be an iteration of the algorithm above. We have

$$\begin{aligned} |C(M^j) \cup \{E_j\}| &\leq \sum_{v \in E_j} \max\{\deg(v) - (b + 1), 0\} + 1 \\ &\leq |E_j| \cdot (\Delta - b) + 1 \\ &\leq l \cdot (\Delta - b) + 1 \end{aligned}$$

We conclude that $1 \geq \frac{|D_{E_j}|}{l(\Delta - b + 1) + 1}$, and hence, D_{E_1}, \dots, D_{E_t} is a partition of \mathcal{E} . It follows

$$|M| = \sum_{i=1}^t 1 \geq \sum_{i=1}^t \frac{|D_{E_i}|}{l(\Delta - b + 1) + 1} = \frac{|\mathcal{E}|}{l(\Delta - b + 1) + 1} .$$

This proves Claim 3. Since $|\mathcal{E}| - \nu_{\mathbf{b}} = \rho_{(\Delta - \mathbf{b})}$, by Claim 3 we get

$$(l(\Delta - b + 1) + 1)|M_{\mathbf{b}}| - \nu_{\mathbf{b}} \geq \rho_{(\Delta - \mathbf{b})} .$$

This implies that $\rho_{(\Delta - \mathbf{b})} \leq l(\Delta - b + 1)\nu_{\mathbf{b}}$, since $\nu_{\mathbf{b}}$ is the cardinality of a maximum \mathbf{b} -matching in \mathcal{H} . This shows that the given transformation (f, g) is an L -reduction from the \mathbf{b} -matching problem in \mathcal{H} to the set \mathbf{d} -multicover problem in \mathcal{H} with a covering factor $\Delta - b$, where $\beta_2 = 1$, $\alpha_2 = l(\Delta - b + 1)$ since b, Δ and l are constants. \square

4.6 Summary and Further Work.

In this chapter we study the (unweighted) maximum \mathbf{b} -matching problem in hypergraphs in the aspect of efficient algorithm design. On the negative side, it is shown that, assuming $\mathcal{P} \neq \mathcal{NP}$, there does not exist polynomial-time approximation scheme for finding

4.6. Summary and Further Work.

maximum \mathbf{b} -matching in l -uniform hypergraphs with degrees bound above by constant $\Delta(\geq 3b)$. Further we present L -reduction from the \mathbf{b} -matching to the set \mathbf{d} -multicover in regular hypergraphs. The \mathbf{b} -matching and set \mathbf{d} -multicover problems under consideration have interesting application in practice. The main results in this chapter are of theoretical interest in algorithm design for both problems.

There remain interesting open questions concerning the \mathbf{b} -matching problem restricted to l -uniform hypergraphs. According to our results, we see that the range $b > \frac{\Delta}{3}$ for the \mathbf{b} -matching problem restricted to l -uniform hypergraphs with respect to the class of complexity MAX-SNP is almost unexplored.

Bibliography

Bibliography

- [1] N. Alon, R. Boppana, J. Spencer. *An asymptotic isoperimetric inequality*. Geometric and Functional Analysis, 8:411-436, 1998.
- [2] N. Alon, D. Moshkovitz and S. Safra. *Algorithmic construction of sets for k -restrictions*. ACM Trans. Algorithms 2:153–177, 2006.
- [3] N. Alon, J. Spencer. *The probabilistic method*. 2nd edition. Wiley Interscience 2000.
- [4] J. Aronson, M. E. Dyer, A. M. Frieze, S. Suen. *Randomized greedy matching II*. Random Structures and Algorithms 6:55-74, 1995.
- [5] S. Arora, C. Lund, R. Motwani, M. Sudan, M. Szegedy. *Proof verification and intractability of approximation problems*. Journal of the ACM, 45:501-555, 1998.
- [6] S. Arora and S. Safra. *Probabilistic checking of proofs: A new characterization of \mathcal{NP}* . Journal of the ACM, 45:70-122, 1998.
- [7] R. Bar-Yehuda and S. Even. *A linear-time approximation algorithm for the weighted vertex cover problem*. Journal of Algorithms, 2:198–203, 1981.
- [8] R. Bar-Yehuda and S. Even. *A local ratio theorem for approximating weighted vertex cover problem*. Annual of Discrete Math., 25:27–46, 1985.

Bibliography

- [9] R. Bar-Yehuda. *Using a homogeneous weights for approximating the partial vertex cover problem*. Proc. Tenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'99), 71–75, 1999.
- [10] C. Berge. *Hypergraphs- Combinatorics of Finite Sets*. North Holland Mathematical Library 1989.
- [11] P. Berman. *A $d/2$ approximation for maximum weight independent set in d -claw free graphs*. Nordic J. of Computing, 7:178–184, 2000.
- [12] N. Bshouty and L. Burroughs. *Massaging a linear programming solution to give 2-approximation for a generalization of the vertex cover problem*. Proc. Annual Symposium on the Theoretical Aspects of Computer Science (STACS'98), 298–308, 1998.
- [13] H. Chernoff. *A measure of the asymptotic efficiency for tests of a hypothesis based on the sum of observables*. Ann. Math. Statist. 23:493–509, 1952.
- [14] V. Chvátal; *A greedy heuristic for the set covering problem*, Math. Oper. Res., 4:233-235, 1979.
- [15] L. Devroye. *Non-Uniform Random Variate Generation*. Springer Verlag, New York, 1986.
- [16] I. Dinur, S. Safra. *On the importance of being biased*. Proc. 34th ACM Symposium on the Theory of Computing (STOC'02), 33–42, 2002.
- [17] R. Duh, M. Fürer. *Approximating k -set cover by semi-local optimization*. Proc. 29th ACM Symposium on the Theory of Computing (STOC'97), 256–264, 1997.

- [18] J. Edmonds. *Maximum matching and polyhedron with (0, 1) vertices*. Journal of Research of National Bureau of Standards, 69:125–130, 1965 .
- [19] M. El Ouali, A. Fretwurst, A. Srivastav. *Inapproximability of b-matching in k-uniform hypergraphs*. Proc. 5th International Workshop on Algorithms and Computation (WALCOM'11), Lecture Note in Computer Science, 57–69, 2011.
- [20] M. El Ouali, H. Fohlin, A. Srivastav. *A randomised approximation algorithm for the partial vertex cover problem in hypergraphs*. First Mediterranean Conference on Algorithms (MedAlg'12), Lecture Note in Computer Science, 174–187, 2012.
- [21] M. El Ouali, H. Fohlin, A. Srivastav. *A randomised approximation algorithm for the hitting set problem*. Proc. 7th International Workshop on Algorithms and Computation, (WALCOM'13), Lecture Note in Computer Science, 101–113, 2013.
- [22] M. El Ouali, G. Jäger. *The b-matching problem in hypergraphs: hardness and approximability*. Proc. of 6th International Conference on Combinatorial Optimization and Applications (COCOA'12), Lecture Note in Computer Science, 200–211, 2012.
- [23] U. Feige. *A threshold of $\ln n$ for approximating set cover*. in: Proc. of the 28th Annual ACM Symposium on the Theory of Computing 1996, (STOC '96), 314–318.
- [24] U. Feige, M. Langberg. *Approximation algorithms for maximization problems arising in graph partitioning*. Journal of Algorithms, 41:174–201, 2001.

Bibliography

- [25] H. Fohlin, L. Kliemann, F. A. Siebert, A. Srivastav, and György Kovács. *Three-dimensional reconstruction of seed implants by randomized rounding and visual evaluation*. Medical Physics 34:967–975, 2007.
- [26] A. Frieze, M. Jerrum. *Improved approximation algorithms for max k -cut and max bisection*. Algorithmica, 18:67–81, 1997.
- [27] Z. Füredi. *Matchings and covers in hypergraphs*. Graphs and Combinatorics 4:115–206, 1988.
- [28] Z. Füredi, J. Kahn, P. D. Seymour. *On the fractional matching polytope of a hypergraph*. Combinatorica 13:167-180, 1993.
- [29] D. Gale, H. W. Kuhn, A. W. Tucker. *Linear programming and theory of games*. In: Activity Analysis of Production and Allocation(T. C. Koopmans, ed), Wiley, New York, 317–329, 1951.
- [30] R. Gandhi, S. Khuller, A. Srinivasan. *Approximation algorithms for partial covering problems*. Proc. International Colloquium on Automata, Languages and Programming (ICALP’01), 225-236, 2001.
- [31] R. Gonen, D. J. Lehmann. *Optimal solutions for multi-unit combinatorial auctions: Branch and bound heuristics*. Proceeding of ACM Conference on Electronic Commerce(EC’00), 13–20, 2000.
- [32] I. Wegener. *complexity theory: exploring the limits of efficient algorithms*. Springer Verlag New York, 2005.
- [33] J. Håstad. *Some optimal inapproximability results*. Journal of the ACM, 48:798–859, 2001.

- [34] M. M. Halldórsson, J. Raghavachari. *Greed is good: Approximating independent sets in sparse and bounded-degree graphs*. Proc. of the 26th Annual ACM Symposium on the Theory of Computing (STOC'94), 439–448, 1994.
- [35] E. Halperin, A. Srinivasan. *Improved approximation algorithms for the partial vertex cover problem*. Proc. International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX'02), 161-174, 2002.
- [36] E. Halperin. *Improved approximation algorithms for the vertex cover problem in graphs and hypergraphs*. Proc. 11th ACM-SIAM Symposium on Discrete Algorithms, 329-337, 2000.
- [37] E. Hazan, S. Safra, O. Schwartz. *On the complexity of approximating k -set packing*. Computational Complexity 15:20-39, 2006.
- [38] N.G. Hall, D.S. Hochbaum. *A fast approximation algorithm for the multicovering problem*. Discrete Applied Mathematics 15:35-40, 1986.
- [39] D. S. Hochbaum. *The t -vertex cover problem: Extending the half integrality framework with budget constraints*. Proc. 1th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, 111–122, 1989.
- [40] D. S. Hochbaum. *Efficient bounds for the stable set, vertex cover and set packing problems*. Discrete Applied Mathematics 6:243-254, 1983.
- [41] D.S. Hochbaum. *Approximation algorithms for the set covering*

Bibliography

- and vertex cover problems*, SIAM J. Computation, 11:555-556, 1982.
- [42] J. Hromkovič. *Algorithmics for hard problems. Introduction to combinatorial optimization, randomization, approximation, and heuristics*. Texts in Theoretical Computer Science. An EATCS Series. Springer Verlag, Berlin, 2001.
- [43] C. A. J. Hurkens, A. Schrijver. *On the size of the systems of sets every t of which have an sdr, with an application to the worst-case ratio of heuristics for packing problems*. SIAM Journal Discrete Math, 2:68-72, 1989.
- [44] G. Jäger, A. Srivastav. *Improved approximation algorithms for maximum graph partitioning problems*. Journal of Combinatorial Optimization 10:133-167, 2005.
- [45] D. S. Johnson. *Approximation algorithms for combinatorial problems*. J. Computer System Sciences, 9:256-278, 1974.
- [46] N. Karmarkar. *A new polynomial-time algorithm for linear programming*. Combinatorica 4:373-395, 1984.
- [47] V. Kann. *Maximum bounded 3-dimensional matching is MAX-SNP-complete*. Inf. Process. Lett. 37:27-35, 1991.
- [48] R. M. Karp. *Reducibility among combinatorial problems*. Complexity of Computer Computations, ed. Raymond E. Miller W. James W. Thatcher, Plenum Press, 85-103, 1972.
- [49] M. Kearns. *The computational complexity of machine learning*. J. Comput. System Sci., 9:257-278, 1990.

- [50] L. Khachiyan. *A polynomial algorithm in linear programming*. Doklady Akademii Nauk SSSR 244:1093–1096, translated in Soviet Mathematics Doklady 20:191–194, 1979.
- [51] S. Khot. *On the power of unique 2-over 1-round games*. Proc. 34th ACM Symposium on Theory of Computing (STOC'02), 767–775, 2002.
- [52] S. Khot and O. Regev. *Vertex cover might be hard to approximate to within 2-epsilon*. J. Comput. Syst. Sci., 74:335–349, 2008.
- [53] S. Khot, N. Bansal. *Inapproximability of hypergraph vertex cover and applications to scheduling problems*. Proc. 37th International Colloquium on Automata, Languages and Programming (ICALP'10), 250 – 261, 2010.
- [54] B. Korte, J. Vygen. *Combinatorial Optimization. Theory and Algorithms*. Algorithms and Combinatorics, 21. Springer Verlag, Berlin, 2000.
- [55] J. Krivelevich. *Approximate set covering in uniform hypergraphs*. J. Algorithms, 25:118–143, 1997.
- [56] P. Krysta. *Greedy approximation via duality for packing, combinatorial auctions and routing*. Proc. 30th International Symposium on Mathematical Foundations of Computer Science (MFCS'05), 615–627, 2005.
- [57] L. Lovász. *On the ratio of optimal integral and fractional covers*. Discrete Math. 13:383–390, 1975.
- [58] C. Lund, M. Yannakakis. *On the hardness of approximating minimization problems*. J. ACM 41:960–981, 1993.

Bibliography

- [59] C. Papadimitriou. *Computational Complexity*. Addison Wesley, 1994.
- [60] C.H. Papadimitriou, M. Yannakakis. *Optimization, approximation, and complexity classes*. Proc. 20th ACM Symposium on Theory of Computing (STOC'88), 229–234, 1988.
- [61] V. T. Paschos. *A survey of approximately optimal solutions to some covering and packing problems*. ACM Computing Surveys 29:171–209, 1997.
- [62] A. B. Marsh. *Matching Algorithms*. Ph. D. thesis, Johns Hopkins University, Baltimore 1979.
- [63] C. McDiarmid. Concentration. In: M. Habib, C. McDiarmid, J. Ramirez-Alfonsin, B. Reed (Eds.), *Probabilistic Methods for Algorithmic Discrete Mathematics*. Springer Verlag Berlin, 195–248, 1998.
- [64] B. Monien and E. Speckenmeyer. *Ramsey numbers and approximation algorithm for the vertex cover problem*. Acta Inform. 22:115–123, 1985
- [65] M. Okun. *On approximation of the vertex cover problem in hypergraphs*. Discrete Optimization 2:101–111, 2005.
- [66] J. Radhakrishnan, A. Ta-Shma. *Bounds for dispersers, extractors, and depth-two superconcentrators*. SIAM Journal on Discrete Mathematics, 13:2–24, 2000.
- [67] P. Raghavan. *Probabilistic construction of deterministic algorithms: approximating packing integer programs*. J. Comput. System Sci. 37:130–143, 1986.

- [68] P. Raghavan, C. D. Thompson. *Randomized rounding: a technique for provably good algorithms and algorithmic proofs*. *Combinatorica*, 7:365–374, 1987.
- [69] D.K. Ray-Chaudhuri. *An algorithm for a minimum cover of an abstract complex*. *Canad. J. Math.* 15:11–14, 1963.
- [70] R. Raz and S. Safra. *A sub-constant error-probability low-degree test, and a sub-constant error-probability PCP characterization of NP*. *Proc. 29th ACM Symposium on Theory of Computing (STOC'97)*, 475–484, 1997.
- [71] V. Rödl, L. Thoma. *Asymptotic packing and the random greedy algorithm*. *Random Structures and Algorithms* 8:161–177, 1996.
- [72] P. Slavík. *Improved Performance of the greedy algorithm for partial cover*, *Information Processing Letters*, 64:251–254, 1997.
- [73] J. Spencer. *Asymptotic packing via a branching process*. *Random Structures and Algorithms* 7:167–172, 1995.
- [74] A. Srinivasan. *Improved approximation guarantees for packing and covering integer programs*. *SIAM Journal on Computing* 29:648–670, 1999.
- [75] A. Srivastav, P. Stangier. *Weighted fractional and integral k -matching in hypergraphs*. *Discrete Appl. Math.* 57:255–269, 1995.
- [76] A. Srivastav, P. Stangier. *Algorithmic Chernoff-Hoeffding inequalities in integer programming*. *Random Structures and Algorithms* 8:27–58, 1996.

Bibliography

- [77] L. Trevisan. *Non-approximability results for optimization problems on bounded degree instances*. Proc. of the 33rd ACM Symposium on Theory of Computing (STOC'01), 453–461, 2001.
- [78] J. von Neumann. *Discussion of maximum problem*. Working paper. Published in: John von Neumann, Collected Works; Vol. VI, Pergamon Press, 27–28, 1963.

Eidesstattliche Erklärung

Ich versichere hiermit, dass die vorliegende Abhandlung, abgesehen von der Beratung durch meinen Betreuer, nach Inhalt und Form meine eigene Arbeit ist. Die Arbeit wurde weder ganz noch in Teilen an anderer Stelle im Rahmen eines Prüfungsverfahrens vorgelegt. Alle inhaltlichen Bestandteile der Arbeit sind zuvor publiziert worden. Die Arbeit ist unter Einhaltung der Regeln guter wissenschaftlicher Praxis der Deutschen Forschungsgemeinschaft entstanden.

Kiel, den 14. Juni 2013

Mourad El Ouali