On Linear and Semidefinite Programming Relaxations for Hypergraph Matching

Yuk Hei Chan Lap Chi Lau*

Department of Computer Science and Engineering
The Chinese University of Hong Kong

Abstract

The hypergraph matching problem is to find a largest collection of disjoint hyperedges in a hypergraph. This is a well-studied problem in combinatorial optimization and graph theory with various applications. The best known approximation algorithms for this problem are all local search algorithms. In this paper we analyze different linear and semidefinite programming relaxations for the hypergraph matching problem, and study their connections to the local search method. Our main results are the following:

- We consider the standard linear programming relaxation of the problem. We provide an algorithmic proof of a result of Füredi, Kahn and Seymour, showing that the integrality gap is exactly $k-1+1/k$ for k-uniform hypergraphs, and is exactly $k-1$ for k-partite hypergraphs. This yields an improved approximation algorithm for the weighted 3-dimensional matching problem. Our algorithm combines the use of the iterative rounding method and the fractional local ratio method, showing a new way to round linear programming solutions for packing problems.
- We study the strengthening of the standard LP relaxation by local constraints. We show that, even after linear number of rounds of the Sherali-Adams lift-and-project procedure on the standard LP relaxation, there are k-uniform hypergraphs with integrality gap at least $k-2$. On the other hand, we prove that for every constant k, there is a strengthening of the standard LP relaxation by only a polynomial number of constraints, with integrality gap at most $(k+1)/2$ for k-uniform hypergraphs. The construction uses a result in extremal combinatorics.
- We consider the standard semidefinite programming relaxation of the problem. We prove that the Lovász ϑ-function provides an SDP relaxation with integrality gap at most $(k+1)/2$. The proof gives an indirect way (not by a rounding algorithm) to bound the ratio between any *local* optimal solution and any optimal SDP solution. This shows a new connection between local search and linear and semidefinite programming relaxations.

1 Introduction

The hypergraph matching problem, also known as the set packing problem, is a fundamental problem in combinatorial optimization with various applications. In general this problem is equivalent to the maximum independent set problem, and is thus hard to approximate [Has99]. In this paper we study the hypergraph matching problem in k-uniform hypergraphs, in which every hyperedge has exactly k vertices; this is also known as the k-set packing problem. This is a generalization of some classical combinatorial optimization problems, e.g. the k-dimensional matching problem [Kann91, HSS06], the maximum independent set problem in bounded degree graphs [BF94, HR97, Tre01], and some graph packing problems [HR06, CMWY08]. This is also an important problem in graph theory [AH00, Aha01], and has interesting connections to the Santa Claus problem [BS06, AFS08] and the partial Latin square problem [GRS03, HJKS07]. All the best known approximation algorithms for the hypergraph matching problem in k-uniform hypergraphs are based on local search methods [HS89, Hal95, CH99, Ber00, BK03b].

Mathematical programming relaxations and local search methods are two important techniques in approximation algorithms, but they appear to be separate techniques with no known direct connections. A topic of recent research is to study the strengthening of linear and semidefinite programming relaxations by local constraints, e.g. Lovász-Schrijver hierarchy, Sherali-Adams hierarchy, Lasserre hierarchy (see [ABL02, Sch08, CMM09, MS09, Tul09] and the references therein). These lift-and-project hierarchies are considered to be a strong computational model which captures many known algorithms. For example, some algorithms obtained by dynamic programming can be captured by the Sherali-Adams hierarchy [BO04, MM09]. Given that the linear programs generated by the Sherali-Adams procedure include all the valid local constraints (see [CMM07] for related work), a natural question is whether they also capture the local search algorithms obtained by changing a few variables (as in [HS89, Hal95, CH99, Ber00, BK03b]). We study this question in the hypergraph matching problem.

*Research is supported by GRF project 413609. Part of this work was done while visiting Microsoft Research New England.

In this paper we analyze the integrality gaps of different linear and semidefinite programming relaxations for the hypergraph matching problem, and study their connections to the local search method. For the standard LP relaxation, we provide an algorithmic proof to obtain a tight analysis for the hypergraph matching problem in k-uniform hypergraphs, giving an improved approximation algorithm for the 3-dimensional matching problem. We then analyze stronger linear and semidefinite programming relaxations, and find some interesting connections to the local search method. On one hand, we show that the local search algorithm is not captured by the Sherali-Adams hierarchy, even after linear number of rounds. On the other hand, extending the analysis of a local search algorithm, we construct a polynomial size linear program with integrality gap a constant factor smaller than the linear programming relaxations generated by the Sherali-Adams hierarchy. Furthermore, the results developed can be used to bound the integrality gap of a semidefinite programming relaxation (the Lovász ϑ-function) for the hypergraph matching problem. This provides a way to bound the ϑ-function indirectly (although we do not know how to round the solutions), by using a connection between the local search method and linear and semidefinite programming relaxations.

We remark that our results (except for 3-dimensional matching) do not improve the approximation guarantees obtained by the local search algorithms, but we believe that they give new insights into the strengths of linear and semidefinite programming relaxations, and also provide new tools and ideas for analysis.

1.1 Our Results. We begin with the standard linear programming formulation for the hypergraph matching problem. In the following we use the notation $x(F)$ to denote $\sum_{e \in F} x_e$ for a subset of hyperedges $F \subseteq E$, and $\delta(v)$ to denote the set of hyperedges incident on a vertex v.

$$\text{(LP)} \qquad \begin{array}{lll} \text{maximize} & x(E) & \\ \text{subject to} & x(\delta(v)) \leq 1 & \forall\, v \in V \\ & x_e \geq 0 & \forall\, e \in E \end{array}$$

We provide an algorithmic proof of a result of Füredi, Kahn and Seymour [FKS93], showing that the integrality gap is exactly $k - 1 + 1/k$ for k-uniform hypergraphs, and is exactly $k - 1$ for k-partite hypergraphs (see Section 2 for definition). The results also hold for weighted problems as in [FKS93]. This yields an improved approximation algorithm for the weighted k-dimensional matching problem (see Section 2 for definition) for $k = 3$. The previous best known approximation for the 3-dimensional matching problem is a $(2+\epsilon)$-approximation algorithm for any $\epsilon > 0$ by Arkin and Hassin [AH97] and Berman [Ber00].

THEOREM 1.1. *There is a polynomial time $(k-1)$-approximation algorithm for the weighted k-dimensional matching problem.*

We then study whether adding local constraints would yield stronger linear programming relaxations. For 3-uniform hypergraphs, the Fano plane as shown in Figure 3(a) in Section 3 is an example with integrality gap 7/3. We show that by adding the local constraint $x(P) \leq 1$ to (LP) for every Fano plane P, the resulting *Fano LP* has an improved integrality gap for the hypergraph matching problem in 3-uniform hypergraphs.

THEOREM 1.2. *The Fano LP for unweighted 3-uniform hypergraphs has integrality gap exactly 2.*

Motivated by Theorem 1.2 and the question stated earlier, we study Sherali-Adams relaxations for the hypergraph matching problem, which can generate all valid local constraints on r hyperedges after r rounds of the lift-and-project procedure [Lau03, CMM09]. In the hypergraph matching problem, after r rounds of the Sherali-Adams lift-and-project procedure, given any subset S of r hyperedges, we will have the constraint $x(S) \leq \textsc{Opt}_S$, where $\textsc{Opt}_S$ is the maximum number of disjoint hyperedges in S. For example, in the hypergraph matching problem in 3-uniform hypergraphs, all the Fano plane constraints will be generated in at most 7 rounds. For the case $k = 2$, Mathieu and Sinclair [MS09] have shown that the Sherali-Adams hierarchy provides a linear programming relaxation with integrality gap at most $1+\frac{1}{r}$ after r rounds, and their result coincides with the approximation guarantee obtained by an r-local optimal solution for the graph matching problem [AH97]. For the hypergraph matching problem in k-uniform hypergraphs, Hurkens and Schrijver [HS89] gave a local search $(\frac{k}{2}+\epsilon)$-approximation algorithm for any $\epsilon > 0$. In contrast to the result of Mathieu and Sinclair [MS09], we show that the local search algorithm is not captured by the linear programming relaxations generated by the Sherali-Adams hierarchy, even after a linear number of rounds.

THEOREM 1.3. *There are k-uniform hypergraphs in which the integrality gap for the Sherali-Adams hierarchy on* (LP) *is at least $k-2$, even after $\Omega(n/k^3)$ rounds where n denotes the number of vertices.*

On the other hand, for every constant k, we can construct a polynomial size linear program for the hypergraph matching problem in k-uniform hypergraphs, with integrality gap smaller than those generated by the Sherali-Adams hierarchy (up to a linear number of rounds) by a constant factor[1]. The proof extends the analysis of the local search algorithm in [HS89], and uses a result in extremal combinatorics.

THEOREM 1.4. *For every constant k, there is a polynomial size linear program for k-uniform hypergraphs with integrality gap at most $\frac{k+1}{2}$ in the unweighted problem.*

Using the results developed, we can show that there is a simple semidefinite program (the Lovász ϑ-function [Lov79, Knu94]) that achieves the same integrality gap for *all* k, not just for constant k as in Theorem 1.4.

THEOREM 1.5. *There is a polynomial size semidefinite program for the hypergraph matching problem, with integrality gap at most $\frac{k+1}{2}$ for k-uniform hypergraphs in the unweighted problem.*

We are not aware of any examples with integrality gap larger than $\Omega(k/\log k)$ implied by the hardness result in [HSS06], for both the LP relaxation in Theorem 1.4 and the SDP relaxation in Theorem 1.5.

1.2 Techniques. The proof of Theorem 1.1 is based on a combination of the iterative rounding method and the fractional local ratio method, showing a new way to round linear programming solutions for packing problems. The standard iterative rounding method [Jain01] is designed for covering problems: if there is a variable with large fractional value, then we can round up the value of this variable to one. By doing so, the covering constraints are still satisfied and thus the process can be iterated. However, for packing problems, even if there is a variable with large fractional value, we could not simply round up the value of this variable to one, because many packing constraints may be violated. Instead of using the fractional values to decide which hyperedges to round up, the idea is to iteratively use the fractional values to define a "good ordering" of the hyperedges (a similar idea is also used in [FKS93]). By using the properties of extreme point solutions, we can define an ordering $\{e_1, e_2, \ldots, e_m\}$ of the hyperedges, so that the total fractional value of the hyperedges in $N[e_i] \cap \{e_i, e_{i+1}, \ldots, e_m\}$ is at most $k-1$, where $N[e_i]$ denotes the set of hyperedges that intersect e_i. Then we can use the fractional local ratio method as in [BHN+06] to obtain an efficient approximation algorithm.

The proofs of Theorem 1.4 and Theorem 1.5 are based on a new connection between the analysis of the local search method, linear programming relaxations, and semidefinite programming relaxations. First we extend the analysis of the local search algorithm in [HS89] to construct an exponential size linear program with integrality gap at most $(k+1)/2$. The proof shows a direct connection between the local search algorithm in [HS89] and the LP relaxation - the ratio between any 2-local optimal solution (see Section 4 for definition) and any optimal solution to the exponential size linear program is at most $(k+1)/2$. To prove Theorem 1.4, we use a result in extremal combinatorics to rewrite the exponential size linear program as a polynomial size linear program, as long as k is a constant. To prove Theorem 1.5, we use known results on Lovász ϑ-function to show that a polynomial size semidefinite program is stronger than the exponential size linear program, and thus has integrality gap at most $\frac{k+1}{2}$. This gives an indirect way to bound the integrality gap of the ϑ-function, without providing a rounding algorithm. Previously either the Sandwich theorem or a rounding algorithm is used to bound the ϑ-function (see below), our results show another way to bound the integrality gap of the ϑ-function.

1.3 Related Work. The hypergraph matching problem in k-uniform hypergraphs is a well-studied problem in combinatorial optimization. For the unweighted problem, Hurkens and Schrijver [HS89] gave a $(\frac{k}{2} + \epsilon)$-approximation algorithm. For the weighted problem, Arkin and Hassin [AH97] gave a $(k - 1 + \epsilon)$-approximation algorithm, Chandra and Halldórsson [CH99] gave a $(\frac{2(k+1)}{3} + \epsilon)$-approximation algorithm, and Berman [Ber00] gave a $(\frac{k+1}{2} + \epsilon)$-approximation algorithm. All the above algorithms are based on local search, and run in polynomial time for any $\epsilon > 0$. On the other hand, Hazan, Safra and Schwartz [HSS06] proved that the problem is hard to approximate within a factor of $O(\frac{k}{\log k})$. For small value of k, Berman and Karpinski [BK03a] obtained a $98/97 - \epsilon$ hardness for 3-dimensional matching (which implies the same hardness for 3-set packing), while Hazan, Safra and Schwartz [HSS03] obtained $54/53 - \epsilon$, $30/29 - \epsilon$ and $23/22 - \epsilon$ hardness for 4, 5 and 6-dimensional matching respectively.

The hypergraph matching problem in k-uniform hypergraphs is also a well-studied problem in graph theory. Ryser conjectured that in a k-partite hypergraph,

[1]It was known that the Sherali-Adams relaxations may not provide the best linear programming relaxations. In the graph matching problem, the linear programs generated by the Sherali-Adams hierarchy are weaker than the Edmonds' linear program. But the Edmonds' linear program is of exponential size (while Theorem 1.4 gives a polynomial size linear program) and also the Sherali-Adams relaxations provide an approximation scheme (while there is a constant factor separation for hypergraph matching).

the ratio between the minimum vertex cover and the maximum matching is at most $k-1$. It has been proved by Aharoni [Aha01] for $k=3$ using a topological method [AH00]. A fractional version is proved by Füredi [Fur81]: he shows that the integrality gap of (LP) is at most $k-1$ whenever the hypergraph does not contain a projective plane of order $k-1$ (see Section 3 for definition) as a subhypergraph, and is at most $k-1+\frac{1}{k}$ for k-uniform hypergraphs. Füredi, Kahn and Seymour [FKS93] extended these results to the weighted case. We remark that the proofs in [Fur81, FKS93] are non-algorithmic, and do not imply Theorem 1.1.

Using lift-and-project methods in approximation algorithms was first studied in the work of Arora, Bollobás and Lovász [ABL02], and since then it has been studied extensively in recent years. The Sherali-Adams hierarchy is known to be stronger than the Lovász-Schrijver linear programming hierarchy, and Lasserre semidefinite programming hierarchy is known to be stronger than the Sherali-Adams hierarchy [Lau03]. Recently strong lower bounds have been obtained for the Sherali-Adams and Lasserre hierarchies for different problems [CMM07, Sch08, CMM09, Tul09]. For the graph matching problem, Mathieu and Sinclair have shown that the integrality gap is at most $1+\frac{1}{r}$ after r rounds of Sherali-Adams relaxations [MS09]. Charikar, Makarychev and Makarychev have shown a connection between integrality gaps for Sherali-Adams relaxations for cut problems to local-global properties in metric spaces [CMM07].

The Lovász ϑ-function is an important technique in estimating the independence number of a graph. It was first introduced by Lovász to bound the Shannon capacity of a graph [Lov79]. In general the integrality gap of the ϑ-function could be very large [Fei95]. It is an interesting open problem whether the ϑ-function provides a better bound for special classes of graphs, and it has been studied for graphs with large independent sets [AK98] and random graphs [FK03, Coja05]. The Sandwich theorem [GLS81, GLS88] shows that the ϑ-function is sandwiched between the independence number and the clique cover number; it can be used to bound the integrality gap of ϑ-function if this ratio is bounded for a certain class of graphs. It is remarkable that this is the only known efficient method to compute the maximum independent set size for perfect graphs [GLS84, GLS88].

2 Integrality Gaps of the Standard Linear Programming Relaxation

In the weighted hypergraph matching problem, we are also given a weight w_e on each hyperedge, and the objective is to find a maximum weighted matching. A hypergraph is called k-partite if the set of vertices can be partitioned into k disjoint sets $V_1, V_2, \ldots, V_k$, and each hyperedge intersects every set of the partition in exactly one vertex. The weighted k-dimensional matching problem is to find a maximum weighted matching in a k-partite hypergraph. When $k=2$ this is the bipartite matching problem. For the analysis of the iterative algorithm, we consider the following more general linear program, denoted by $LP(G,\mathcal{B})$, where $\mathcal{B}$ denotes the vector of all degree bounds $0 \le B_v \le 1$ for each vertex $v \in V$. Initially $B_v = 1$ for each $v \in V$.

$$
\begin{aligned}
\text{maximize} \quad & \sum_{e\in E} w_e\, x_e \\
\text{subject to} \quad & x(\delta(v)) \le B_v \qquad \forall\, v \in V \\
& x_e \ge 0 \qquad \forall\, e \in E
\end{aligned}
$$

Our rounding algorithm is based on the following properties of the extreme point solutions to $LP(G,\mathcal{B})$. Each constraint $x(\delta(v)) \le B_v$ defines a vector in $\mathbb{R}^{|E|}$: the vector has a 1 corresponding to each hyperedge $e \in \delta(v)$ and 0 otherwise. We call this vector the characteristic vector of $\delta(v)$, and denote it by $\chi_{\delta(v)}$.

LEMMA 2.1. *Given any extreme point solution x to linear program $LP(G,\mathcal{B})$ such that $x_e > 0$ for each $e \in E$ there exists $W \subseteq V$ such that*

1. $x(\delta(v)) = B_v > 0$ *for each* $v \in W$.
2. *The characteristic vectors* $\{\chi_{\delta(v)} : v \in W\}$ *are linearly independent.*
3. $|W| = |E|$.

Proof. In an extreme point solution x of an LP, it is known that the number of non-zero variables is at most the number of linearly independent tight constraints (constraints that achieve equality); this holds for any LP. Since we assume that $x_e > 0$ for every hyperedge e, there will be no tight constraints of the form $x_e \ge 0$. And so the only tight constraints come from the degree constraints $x(\delta(v)) = B_v$. Let W be the set of vertices where the degree constraints are tight and linearly independent, i.e. $x(\delta(v)) = B_v$ for every $v \in W$ and no constraint in W can be written as a linear combination of other constraints in W. Then conclusion 1 holds by the definition of W, and the condition that every hyperedge has fractional value $x_e > 0$. Conclusion 2 follows from the definition of W. Conclusion 3 follows from the above property of an extreme point solution, and the condition that every hyperedge has a non-zero fractional value. $\square$

Our algorithm consists of two phases. In the first phase we use an iterative algorithm to provide a "good"

Iterative k-Dimensional Matching Algorithm

1. Find an optimal extreme point solution $\mathbf{x}$ to $LP(G,\mathcal{B})$ where $B_v = 1$ for all v. Remove every hyperedge e with $x_e = 0$. Initialize $F \leftarrow \emptyset$.
2. For i from 1 to $|E(G)|$ do
   (a) Find a hyperedge e with $x(N[e]) \leq k-1$.
   (b) Set $e_i \leftarrow e$ and $F \leftarrow F \cup \{e_i\}$.
   (c) Remove e from G.
   (d) Decrease B_v by x_e for all $v \in e$.
3. $M \leftarrow$ **Local-Ratio**$(F, \mathbf{w})$, where $\mathbf{w}$ is the weight vector of the hyperedges.
4. Return M.

Figure 1: k-Dimensional Matching Algorithm

Local-Ratio$(F, \mathbf{w})$

1. Remove from F all hyperedges with non-positive weights.
2. If $F = \emptyset$, then **return** $\emptyset$.
3. Choose from F the hyperedge e with the smallest index. Decompose the weight vector $\mathbf{w} = \mathbf{w}_1 + \mathbf{w}_2$ where

$$w_1(e') = \begin{cases} w(e) & \text{if } e' \in N[e], \\ 0 & \text{otherwise.} \end{cases}$$

4. $M' \leftarrow$ Local-Ratio$(F, \mathbf{w}_2)$.
5. If $M' \cup \{e\}$ is a matching, **return** $M' \cup \{e\}$; else **return** M'.

Figure 2: The local ratio routine

ordering of the hyperedges. In the second phase we apply the local ratio method to this good ordering to obtain a matching with weight at least $\frac{1}{k-1}$ the optimal. In the following let $N[e]$ be the set of hyperedges that intersect the hyperedge e; note that $e \in N[e]$.

To prove the correctness of the algorithm, we show that the iterative algorithm always succeed in finding an ordering with a good property. Then, using the property of the ordering, we prove that the local ratio method will return a matching with cost at least $\frac{1}{k-1}$ the optimum. First we prove that the iterative algorithm will always succeed in finding a good ordering.

Theorem 2.2. *In the k-dimensional matching problem, the iterative algorithm in Figure 1 will always succeed in finding an ordering of the hyperedges with $x(N[e_i] \cap \{e_i, e_{i+1}, \ldots, e_m\}) \leq k-1$ for all $1 \leq i \leq m$, where m is the number of hyperedges in $\mathbf{x}$ with positive fractional value.*

The proof of Theorem 2.2 consists of two steps. First, in Lemma 2.3, we prove that there is a hyperedge e with $x(N[e]) \leq k-1$ in an extreme point solution to $LP(G,\mathcal{B})$. Since the initial solution $\mathbf{x}$ is an extreme point solution, this implies that the first iteration of Step 2 of the iterative algorithm will succeed. Then we prove in Lemma 2.4 that the remaining solution (after removing e and updating B_v) is still an extreme point solution to $LP(G,\mathcal{B})$. Therefore, by applying Lemma 2.3 inductively, the iterative algorithm will succeed in finding an ordering of hyperedges $\{e_1, \ldots, e_m\}$ with $x(N[e_i] \cap \{e_i, e_{i+1}, \ldots, e_m\}) \leq k-1$ for all $1 \leq i \leq m$. Now we prove Lemma 2.3.

Lemma 2.3. *Let $\mathbf{x}$ be an extreme point solution to $LP(G,\mathcal{B})$ for the k-dimensional matching problem. If $x_e > 0$ for all $e \in E$, then there is a hyperedge e with $x(N[e]) \leq k-1$.*

Proof. Let W be the set of vertices with tight degree constraints as described in Lemma 2.1. To show that there is a hyperedge with the required property, we first prove that in any extreme point solution of $LP(G,\mathcal{B})$ there is a vertex in W of degree at most $k-1$. Suppose, by way of contradiction, that every vertex in W is of degree at least k. This implies that

$$|W| = |E| = \frac{\sum_{v \in V} |\delta(v)|}{k} \geq \frac{\sum_{v \in W} |\delta(v)|}{k} \geq |W|,$$

where the first equality follows from Lemma 2.1, the second equality follows because every hyperedge contains exactly k vertices, and the last inequality follows because every vertex in W is of degree at least k. Hence equality must hold everywhere. Thus the first inequality implies that every hyperedge is contained in W. Let $V_1, V_2, \ldots, V_k$ be the k-partition of V, and $W_i = W \cap V_i$ for $1 \leq i \leq k$. Since each hyperedge intersects W_i exactly once, we have

$$\sum_{v \in W_1} \chi_{\delta(v)} = \sum_{v \in W_2} \chi_{\delta(v)}.$$

This implies that the characteristic vectors in W are not linearly independent, contradicting to Lemma 2.1. Therefore there is a vertex $v_1 \in W$ of degree at most $k-1$. Let $e = \{v_1, v_2, \ldots, v_k\}$ be the hyperedge in $\delta(v_1)$

with the largest fractional value. Since v_1 is of degree at most $k-1$, this implies that $(k-1)x_e \geq x(\delta(v_1))$. Therefore,

$$\begin{aligned} x(N[e]) &\leq \sum_{i=1}^{k} x(\delta(v_i)) - (k-1)x_e \\ &\leq \sum_{i=2}^{k} x(\delta(v_i)) \\ &\leq \sum_{i=2}^{k} B_i \\ &\leq k-1. \qquad \square \end{aligned}$$

The following lemma allows Lemma 2.3 to be applied inductively to complete the proof of Theorem 2.2.

LEMMA 2.4. *In any iteration of Step 2 of the algorithm in Figure 1, the fractional solution is an extreme point solution to $LP(G, \mathcal{B})$.*

Proof. Suppose the graph in the current iteration is $G = (V, E)$. Let $\mathbf{x}_E$ be the restriction of the initial extreme point solution $\mathbf{x}$ to E. We prove by induction on the number of iterations that $\mathbf{x}_E$ is an extreme point feasible solution to $LP(G, \mathcal{B})$. This holds in the first iteration by Step 1 of the algorithm. Let $e = \{v_1, v_2, \ldots, v_k\}$ be the hyperedge found in Step 2(a) of the algorithm. Let $E' = E - e$ and $G' = (V, E')$. Let $\mathcal{B}'$ be the updated degree bound vector. We prove that $\mathbf{x}_{E'}$ is an extreme point feasible solution to $LP(G', \mathcal{B}')$. Since the degree bounds of v_1, v_2, v_3 are decreased by exactly x_e, it follows that $\mathbf{x}_{E'}$ is still a feasible solution. Suppose, to the contrary, that $\mathbf{x}_{E'}$ is not an extreme point solution to $LP(G', \mathcal{B}')$. This means that $\mathbf{x}_{E'}$ can be written as a convex combination of two different feasible solutions $\mathbf{y}_1$ and $\mathbf{y}_2$ to $LP(G', \mathcal{B}')$. Extending $\mathbf{y}_1$ and $\mathbf{y}_2$ by setting the fractional value on e to be x_e, this implies that $\mathbf{x}_E$ can be written as a convex combination of two different feasible solutions to $LP(G, \mathcal{B})$, contradicting that $\mathbf{x}_E$ is an extreme point solution. Hence $\mathbf{x}_{E'}$ is an extreme point solution to $LP(G', \mathcal{B}')$. $\square$

To provide an efficient rounding algorithm, we use the fractional local ratio method as in [BHN+06]. The following is the basic result of the local ratio method.

THEOREM 2.5. ([BBFR04]) *Let $\mathcal{C}$ be a set of vectors in $\mathbb{R}^n$. Let $\mathbf{w}, \mathbf{w}_1, \mathbf{w}_2 \in \mathbb{R}^n$ be such that $\mathbf{w} = \mathbf{w}_1 + \mathbf{w}_2$. Suppose $\mathbf{x} \in \mathcal{C}$ is r-approximate with respect to $\mathbf{w}_1$ and r-approximate with respect to $\mathbf{w}_2$. Then $\mathbf{x}$ is r-approximate with respect to $\mathbf{w}$.*

Using the ordering in Theorem 2.2, we prove the performance guarantee of the approximation algorithm in Figure 2. Note that by construction the local ratio routine returns a matching. It remains to prove that the cost of the returned matching is at least $\frac{1}{k-1}$ of the optimum. The following result implies Theorem 1.1.

THEOREM 2.6. *Let $\mathbf{x}$ be an optimal solution to $LP(G, \mathcal{B})$ for the k-dimensional matching problem. The matching M returned by the algorithm in Figure 2 satisfies $w(M) \geq \frac{1}{k-1} \cdot \mathbf{w} \cdot \mathbf{x}$.*

Proof. The proof is by induction on the number of hyperedges having positive weights. The theorem holds in the base case when there are no hyperedges with positive weights. Let e be the hyperedge e chosen in Step 3 of the algorithm. Since e has the smallest index in the ordering, by Theorem 2.2, we have $x(N[e]) \leq k-1$. Let $\mathbf{w}, \mathbf{w}_1, \mathbf{w}_2$ be the weight vectors computed in Step 3 of the algorithm. Let $\mathbf{y}'$ and $\mathbf{y}$ be the characteristic vectors for M' and M obtained in Step 4 and Step 5 respectively. Since $w(e) > 0$ and $w_2(e) = 0$, $\mathbf{w}_2$ has fewer hyperedges with positive weights than $\mathbf{w}$. By induction, $\mathbf{w}_2 \cdot \mathbf{y}' \geq \frac{1}{k-1} \cdot \mathbf{w}_2 \cdot \mathbf{x}$. Since $w_2(e) = 0$, this implies that $\mathbf{w}_2 \cdot \mathbf{y} \geq \frac{1}{k-1} \cdot \mathbf{w}_2 \cdot \mathbf{x}$. By Step 5 of the algorithm, at least one hyperedge in $N[e]$ is in M. Since $x(N[e]) \leq k-1$ and $w_1(e') = w(e)$ for all $e' \in N[e]$, it follows that $\mathbf{w}_1 \cdot \mathbf{y} \geq \frac{1}{k-1} \cdot \mathbf{w}_1 \cdot \mathbf{x}$. Therefore, by Theorem 2.5, we have $\mathbf{w} \cdot \mathbf{y} \geq \frac{1}{k-1} \cdot \mathbf{w} \cdot \mathbf{x}$. This shows that M is a $(k-1)$-approximate solution to the k-dimensional matching problem. $\square$

This completes the proof of Theorem 1.1. The same techniques can be used to prove that the integrality gap of (LP) is exactly $k - 1 + \frac{1}{k}$ for k-uniform hypergraphs. The analysis is tight, as there are examples of k-uniform hypergraphs having this integrality gap; see Section 3. Here we sketch the proof for k-uniform hypergraphs and omit the details. Using the proof of Lemma 2.3, we can show that there is a vertex v with degree at most k, and this implies that there is a hyperedge incident on v with $x(N[e]) \leq k - 1 + \frac{1}{k}$. Then the same local ratio method implies that the integrality gap is at most $k - 1 + \frac{1}{k}$.

3 Linear Programming Relaxations with Local Constraints

In this section we study the strengthenings of (LP) by local constraints. Before that we first see the integrality gap example for (LP) [Fur81]. Consider a projective plane of order $k-1$, which is a hypergraph H with the following properties: (1) it is k-uniform (every hyperedge is of size k), (2) it is k-regular (every vertex is of degree k), (3) it is intersecting (every pair of hyperedges intersect), (4) it has $k^2 - k + 1$ hyperedges. It is known that a projective plane of order q exists if q is a prime power (see e.g. [MN08]); see Figure 3(a) for the projective plane of order 2. Since H is intersecting, the maximum matching size is one. On the other hand, since H is k-regular, by setting $x_e = \frac{1}{k}$ for each hyperedge e, this is a feasible solution to (LP). Since it has $k^2 - k + 1$ hyperedges, the integrality gap for (LP) is thus $k - 1 + \frac{1}{k}$.

3.1 Fano Plane Constraint for 3-Uniform Hypergraphs. In this section, we show that by adding additional constraints for the Fano planes to (LP), we can improve the integrality gap for the hypergraph matching problem in 3-uniform hypergraphs from $\frac{7}{3}$ to 2. For every seven hyperedges that form a Fano plane P, the *Fano plane constraint* states that the sum of fractional values in this seven hyperedges must not exceed two[2]:

$$x(P) \leq 2 \qquad \forall \text{ Fano plane } P.$$

We call the resulting linear program the Fano linear program, denoted by *Fano-LP*.

We consider a counterexample H to Theorem 1.2 with the minimum number of hyperedges. The major step is to prove that there is no Fano plane (as a subhypergraph) in any extreme point solution to H. Then the following result of Füredi [Fur81] shows that such a minimal counterexample does not exist, and thus Theorem 1.2 follows. We leave the details to the full version of this paper.

Theorem 3.1. ([Fur81]) *If H is a 3-uniform hypergraph which does not contain a Fano plane, then the integrality gap of* (LP) *for the hypergraph matching problem is at most two.*

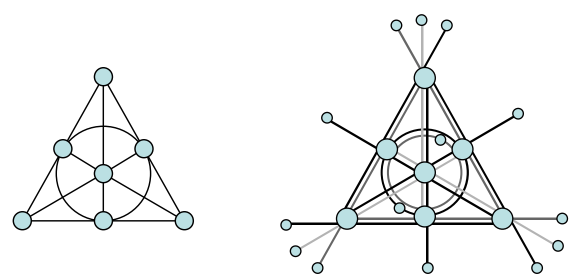
3.2 Sherali-Adams Relaxations. We then study the integrality gap of Sherali-Adams relaxations on (LP), which can generate all valid local constraints on r hyperedges after r rounds of lift-and-project procedure [Lau03, CMM09]. To write the l-round Sherali-Adams relaxation for (LP), for each original constraint $\sum_{e\in\delta(v)} x_e - 1 \leq 0$, we have the following constraint for each pair of disjoint subsets I, J of E with $|I \cup J| \leq l$:

$$\Big(\sum_{e\in\delta(v)} x_e - 1\Big)\prod_{i\in I} x_i \prod_{j\in J}(1-x_j) \leq 0. \tag{3.1}$$

Also we have the constraint

$$\prod_{i\in I} x_i \prod_{j\in J}(1-x_j) \geq 0$$

for each pair of disjoint subsets I, J of E with $|I\cup J| \leq l$. We then expand these polynomial constraints, replace each square term x_e^2 by x_e, and replace each monomial $\prod_{i\in S} x_i$ by a variable y_S for each subset $S \subseteq E$, to obtain a linear program in the y variables. The objective function of this linear program is to maximize $\sum_{e\in E} y_{\{e\}}$.



(a) Fano plane. (b) Gap example.

Figure 3: (a) The Fano plane is an intersecting hypergraph with 7 hyperedges of size 3. (b) The integrality gap example with $k = 4$ and $q = 2$.

We construct the integrality gap example for a k-uniform hypergraph as follows. Take a projective plane P of order $k-2$, we construct a hypergraph H as follows: for each hyperedge e of size $k-1$ in P, we have q hyperedges $\{e \cup v_1^e, e \cup v_2^e, \ldots, e \cup v_q^e\}$ of size k in H, where $v_1^e, v_2^e, \ldots, v_q^e$ are new vertices of degree 1. Since P is $(k-1)$-uniform and has $k^2 - 3k + 3$ vertices and $k^2 - 3k + 3$ hyperedges of size $k - 1$, the hypergraph H is k uniform and has $(k^2 - 3k + 3)(q + 1)$ vertices and $(k^2 - 3k + 3)q$ hyperedges. The degree of each vertex in P is exactly $(k - 1)q$. See Figure 3(b) for an example with $k = 4$ and $q = 2$. From the construction, since P is intersecting, H is also intersecting, and thus the maximum matching size for H is one. Using the special structure of H, the following lemma follows from the results of Mathieu and Sinclair [MS09].

Lemma 3.2. *In the l-round Sherali-Adams relaxation for the hypergraph matching problem in H, any feasible solution must have $y_S = 0$ for all S with $|S| \geq 2$.*

Proof. This proof is essentially the same as the proof of Lemma 3.2 in Mathieu and Sinclair [MS09]. Let $S = \{e_1, e_2, \ldots, e_j\}$ for some $j \geq 2$. First we show that $y_S \geq 0$. Take the constraint $x_{e_1} \geq 0$ and multiply it by $\prod_{i=2}^{j} x_{e_i} \geq 0$, this will give $y_S = \prod_{i=1}^{j} x_{e_i} \geq 0$. In fact it is true that $y_T \geq 0$ for every non-empty subset $T \subseteq E$.

Now we show that $y_S = 0$. Since H is intersecting and $|S| \geq 2$, there are two hyperedges sharing a vertex. Let $e_1 = \{u, w_1\}$ and $e_2 = \{u, w_2\}$. Consider the

[2] Actually we can write the stronger constraint $x(P) \leq 1$ for each Fano plane, but for our analysis we need to use the weaker constraints.

following constraint in the Sherali-Adams relaxation:

$$\Big(\sum_{e\in\delta(u)} x_e - 1\Big)\prod_{i=2}^{j} x_{e_i} \le 0.$$

Expanding it and replacing x_e^2 by x_e, the term $-\prod_{i=2}^{j} x_{e_i}$ is cancelled by the term $x_{u,w_2}\prod_{i=2}^{j} x_{e_i}$ since $x_{u,w_2}x_{e_2} = x_{e_2}$. Therefore the constraint becomes a summation of monomials, all of them having coefficient $+1$. Since $y_T \ge 0$ for all T, all the remaining monomial y_T in this constraint must have $y_T = 0$. In particular, the term $y_S = \prod_{i=1}^{j} x_{e_i}$ appears in this constraint, and thus $y_S = 0$. □

Now we show that the l-round Sherali-Adams relaxation for H still has a large fractional solution. With the lemma the constraints for the Sherali-Adams relaxation become very simple. For each constraint with $|I| \ge 2$, all the terms on the left hand side of (3.1) are equal to zero, and thus the constraint becomes trivial. For each constraint with $|I| = 1$, the constraint reduces to the constraint that $y_{\{e\}} \ge 0$ for some $e \in E$. If $|I| = 0$, then the constraint will become

$$\sum_{e\in\delta(v)} y_{\{e\}} + \sum_{j\in J} y_{\{j\}} \le 1.$$

Similarly constraints of the form $\prod_{i\in I} x_i \prod_{j\in J}(1-x_j) \ge 0$ will become $\sum_{j\in J} y_{\{j\}} \le 1$. Since $|J| \le l$, by setting the fractional value of each hyperedge to be $\frac{1}{(k-1)q+l}$, all the constraints in the Sherali-Adams relaxation will be satisfied. The objective value of this fractional solution is equal to

$$\frac{(k^2-3k+3)q}{(k-1)q+l}.$$

For the l-round Sherali-Adams relaxation, the integrality gap is smaller than $k-2$ only when $l > \frac{q}{k-2}$. Consider the case when k is a constant and q is much larger than k. Then the Sherali-Adams hierarchy will require $\Omega(|V(H)|)$ number of rounds to generate a linear programming relaxation with integrality gap smaller than $k-2$ for H. This proves Theorem 1.3. Finally, we remark that Theorem 1.3 also holds for k-partite hypergraphs (without projective plane as a subhypergraph), by replacing the projective plane by a *truncated* projective plane (see e.g. [MN08]); we skip the details.

4 Stronger Linear and Semidefinite Programming Relaxations

In this section we first show that an exponential size linear program has integrality gap at most $\frac{k+1}{2}$. Then we show how to construct a polynomial size linear program with the same integrality gap, and a semidefinite program with probably smaller integrality gap. We remark that the linear program is of polynomial size only when k is a constant, but the semidefinite program is of polynomial size for all k.

We note that the integrality gap example for the Sherali-Adams relaxations in Section 3 actually have matching size only one. Call a set K of hyperedges an intersecting family if every two hyperedges in K have a non-empty intersection. We consider a strengthening of (LP) by adding the following constraint for each intersecting family K:

$$x(K) \le 1.$$

Call the resulting linear program the intersecting family linear program. In general this linear program has exponentially many constraints, and is NP-hard to check whether a fractional solution is a feasible solution to this linear program (i.e. no polynomial time separation oracle). Nevertheless, extending the analysis of a local search algorithm in [HS89], we can show that the integrality gap of the intersecting family linear program is at most $\frac{k+1}{2}$. In particular, the proof directly compares an optimal fractional solution to a *2-local* optimal integral solution, where a 2-local optimal solution is an integral solution where we cannot increase the size of the matching by removing at most one hyperedge and adding at most two hyperedges (an optimal integral solution is clearly a 2-local optimal solution). However the proof does not provide a rounding algorithm.

Theorem 4.1. *The ratio between any LP solution to the intersecting family LP and any 2-local optimal solution is at most $\frac{k+1}{2}$. Thus the integrality gap of the intersecting family LP is at most $\frac{k+1}{2}$.*

Proof. Let M be a 2-local optimal matching. Let x be a feasible solution to the intersecting family LP, and let F be the set of hyperedges with $x_e > 0$. To prove the lemma we prove that $x(F) \le (k+1)|M|/2$. We let F_1 to be the subset of F in which every hyperedge in F_1 intersect at most one hyperedge in M, and let F_2 to be the subset of F in which every hyperedge in F_2 intersect at least two hyperedges in M. Note that each hyperedge e in F must intersect at least one hyperedge in M; otherwise M is not a 2-local optimal matching because $M + e$ is a larger matching. Consider a hyperedge e in M. We claim that $F_1(e) := \{f \mid f \in F_1 \text{ and } f \cap e \neq \emptyset\}$ is an intersecting family. Suppose otherwise, then there are two disjoint hyperedges f_1, f_2 in F_1. Since $f_1, f_2 \in F_1$, they do not intersect other hyperedges in M. Hence $M - e + f_1 + f_2$ is a larger matching than M, contradicting that M is a 2-local optimal matching. Therefore $F_1(e)$ is an intersecting family. So, by the

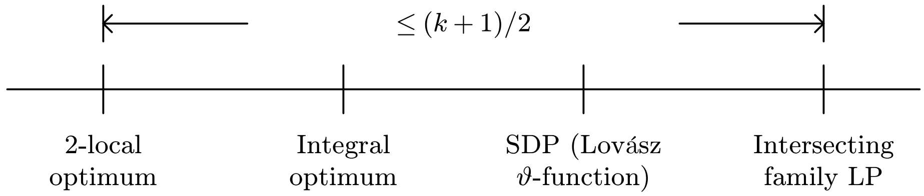


Figure 4: Relations between various solutions to the k-set packing problem.

intersecting family constraint, $x(F_1(e)) \leq 1$, and hence $x(F_1) \leq |M|$. There are $k|M|$ vertices in M since each hyperedge is of size k. By the degree constraints (a special case of intersecting family constraints), we have $x(F_2) \leq k|M| - x(F_1)$. In fact, since each hyperedge in F_2 intersects at least two hyperedges in M, we have $x(F_2) \leq (k|M| - x(F_1))/2$. Therefore the lemma follows:

$$\begin{aligned} x(F) &= x(F_1) + x(F_2) \\ &\leq x(F_1) + \frac{k|M| - x(F_1)}{2} \\ &= \frac{k|M| + x(F_1)}{2} \\ &\leq \frac{(k+1)|M|}{2}. \end{aligned} \qquad \square$$

4.1 Linear Programming Relaxation. In the following we show how to rewrite the intersecting family linear program by using only a polynomial number of constraints, as long as k is a constant (the polynomial grows exponentially in k). Observe that in the example in Section 3, although the number of vertices and hyperedges in the intersecting family is large, all the intersections take place in a small number of vertices (in the projective plane). We will define the concept of kernel as follows. Let $S \subseteq V$ be a subset of vertices. For each hyperedge e, we define $e_S := e \cap S$, and for a subset K of hyperedges, we define $K_S = \{e_S \mid e \in K\}$. Given an intersecting family K, we say S is a kernel of K, if K_S is an intersecting family. In the example in Section 3, the projective plane of order $k-2$ is a kernel with a small number of vertices. The following result from extremal combinatorics states that every intersecting family has a small kernel [Cal64]. The point is that the size of the kernel is a function of k independent on the number of vertices; the current best bounds [EL73, Tuza85, Tuza87] show that $f(k) = \Theta(\binom{2k}{k})$.

Theorem 4.2. ([Cal64]) *For every k there exists an $f(k)$ such that for every k-uniform intersecting family K there is a kernel S of cardinality at most $f(k)$.*

Now we show how to use this result to add only a polynomial number of constraints so that each intersecting family has total fractional value at most one. For each subset $S \subseteq V$, we create a new variable x_S if S is a subset of a hyperedge $e \in E(H)$. We add the constraint $x_S = \sum_{e: S \subseteq e} x_e$ so that x_S represents the total fractional value of the number of hyperedges containing S. To enforce that the intersecting family constraints, we enumerate all possible subset of vertices S of size up to $f(k)$, and for each subset S we enumerate all possible intersecting family K within S formed by the new variables (not necessarily subsets of size k, but subsets of size at most k), and write the following kernel constraint:

$$\sum_{S \in K} x_S \leq 1.$$

There are $\sum_{i=1}^{f(k)} \binom{n}{i} \leq n^{f(k)+1}$ possible kernels. For each kernel S of size l, there are at most 2^l new variables (the number of all possible subsets of S) containing in S, and thus there are at most 2^{2^l} intersecting families (the number of all possible hypergraphs in S) induced in S. There is one constraint for each such intersecting family, and so there are at most $n^{f(k)+1} \cdot 2^{2^{f(k)}}$ kernel constraints. Therefore, for every constant k, there are at most a polynomial number of kernel constraints. It follows from Theorem 4.2 that each intersecting family constraint has a corresponding kernel constraint, and thus each intersecting family has total fractional value at most one. Therefore, the intersecting family linear program can be rewritten as a polynomial size kernel linear program for any constant k. This proves Theorem 1.4.

4.2 Semidefinite Programming Relaxation. In the following we show that the Lovász ϑ-function captures all the intersecting family constraints, and thus provides a polynomial size semidefinite program with integrality gap at most $\frac{k+1}{2}$. We remark that the proof follows directly from known results about the ϑ-function [GLS88, Knu94], but it seems that it is the first use of these results to give a nontrivial bound on the integrality gap of the ϑ-function.

To see the connection it is more convenient to view the hypergraph matching problem as an independent set problem. For any hypergraph H, we construct a graph G where each vertex in G represents a hyperedge in H and two vertices in G have an edge if and only if the corresponding hyperedges in H intersect. It follows that H has a matching size of size l if and only if G has an independent set of size l. Also, the intersecting family linear program for hypergraph matchings in H becomes the clique linear program for independent sets in G, in which there is a constraint $\sum_{v \in C} x_v \leq 1$ for each clique C in G. The clique linear program is known as QSTAB(G) in the literature [GLS88, Knu94], and Padberg has shown that the clique constraints define facets for the hypergraph matching problem [Pad73]. The Lovász ϑ-function is defined as follows:

$$
\begin{aligned}
\text{(TH)} \quad \theta(G) \; = \; & \max \sum_{i \in V} x_i \\
\text{s.t.} \quad & \sum_{i \in V} (\mathbf{c}^T \mathbf{u}_i)^2 x_i \leq 1, \qquad \forall \mathbf{c}, \forall\, \mathsf{ONR}\{\mathbf{u}_i\} \\
& x_i \geq 0, \qquad \forall i \in V
\end{aligned}
$$

where $\mathbf{c}$ ranges over all possible unit vectors, and $\{\mathbf{u}_i\}$ ranges over all possible orthonormal representations[3] (ONR) of G. It is known that TH(G) $\subseteq$ QSTAB(G) [GLS88, Knu94]; that is, the ϑ-function is a stronger relaxation than the clique linear program.

Lemma 4.3. ([GLS88]) *Any feasible solution to the ϑ-LP* (TH) *is a feasible solution to the clique linear program.*

Proof. This proof is from [GLS88]. To prove this lemma, we prove that every constraint of the clique linear program is a constraint in the ϑ-LP (TH). For any clique C of G, we define the following orthonormal representation of G. Let I_j be the j-th row of the $n \times n$ identity matrix I. Then

$$
\mathbf{u}_i = \begin{cases} I_1, & \text{if } i \in C, \\ I_i, & \text{otherwise.} \end{cases}
$$

Note that this is indeed an orthonormal representation of G, because if $(i,j) \notin E$, then $\mathbf{u}_i \cdot \mathbf{u}_j = 0$, and thus orthogonal. We set $\mathbf{c} = I_1$. Hence if $i \notin C$, then $\mathbf{c}^T \mathbf{u}_i = 0$, and if $i \in C$, then $\mathbf{c}^T \mathbf{u}_i = 1$. Thus

$$
\sum_{i \in V} (\mathbf{c}^T \mathbf{u}_i)^2 x_i = \sum_{i \in C} x_i.
$$

Therefore the clique constraint is present in the ϑ-LP (TH), and so the lemma follows. $\square$

[3] An orthonormal representation (ONR) of a graph G is a system $(\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n)$ of unit vectors in an Euclidean space such that if $(i,j) \notin E(G)$ then $\mathbf{v}_i$ and $\mathbf{v}_j$ are orthogonal.

It is also known that the ϑ-function is equivalent to the following semidefinite program, which is called the "third" face of the ϑ-function in [GLS88, LS91, Knu94].

$$
\begin{aligned}
(\text{TH}_3) \qquad \theta_3(G) \; = \; & \max \sum_{i,j \in V} \mathbf{w}_i \cdot \mathbf{w}_j \\
\text{s.t.} \quad & \mathbf{w}_i \cdot \mathbf{w}_j = 0, \qquad \forall (i,j) \in E(G) \\
& \sum_{i=1}^{n} \mathbf{w}_i^2 = 1, \\
& \mathbf{w}_i \in \mathbb{R}^n, \qquad \forall i \in V
\end{aligned}
$$

Therefore, by Theorem 4.1 and the above known results in ϑ-function, it follows that there is a polynomial size semidefinite program for hypergraph matching with integrality gap at most $\frac{k+1}{2}$ for k-uniform hypergraphs. This proves Theorem 1.5.

Concluding Remarks

In this paper we analyze different linear and semidefinite programming relaxations for the hypergraph matching problem. Our results show a new connection between the local search method and linear and semidefinite programming relaxations. Also they show that the SDP relaxation is strictly stronger than the LP relaxations. We believe that further investigations of the SDP relaxation is a promising avenue to improve the approximation guarantees obtained by the local search algorithms. As mentioned earlier, we are not aware of any example with integrality gap at least $\Omega(k/\log k)$ as implied by the hardness result in [HSS06]. It would be interesting to obtain a rounding algorithm for the SDP relaxation.

Acknowledgment

We thank Nick Harvey and Mohit Singh for many useful discussions and comments. The research is supported by GRF project 413609.

References

[Aha01] R. Aharoni: *Ryser's Conjecture for Tripartite 3-graphs.* Combinatorica **21**(1), 1–4, 2001.

[AH00] R. Aharoni, P. Haxell: *Hall's Theorem for Hypergraphs.* Journal of Graph Theory **35**, 83–88, 2000.

[AK98] N. Alon, N. Kahale: *Approximating the Independence Number via the ϑ-function.* Mathematical Programming **80**(3), 253–264, 1998.

[AH97] E.M. Arkin, R. Hassin: *On Local Search for Weighted k-Set Packing.* Mathematics of Operations Research **23**(3), 640–648, 1998.

[ABL02] S. Arora, B. Bollobás, L. Lovász: *Proving Integrality Gaps without Knowing the Linear Program.* Proceedings of the 43rd Annual Symposium on Foundations of Computer Science (FOCS), 313–322, 2002.

[AFS08] A. Asadpour, U. Feige, A. Saberi: *Santa Claus Meets Hypergraph Matchings.* Proceedings of APPROX 2008, 10–20, 2008.

[BBFR04] R. Bar-Yehuda, K. Bendel, A. Freund, D. Rawitz: *Local Ratio: A Unified Framework for Approximation Algorithms. In Memoriam: Shimon Even 1935–2004.* ACM Computing Surveys **36**(4), 422–463. 2004.

[BHN+06] R. Bar-Yehuda, M.M. Halldórsson, J.(S.) Naor, H. Shachnai, I. Shapira: *Scheduling Split Intervals.* SIAM Journal on Computing **36**(1), 1–15, 2006.

[BS06] N. Bansal, M. Sviridenko: *The Santa Claus Problem.* Proceedings of the 38th Annual ACM Symposium on Theory of Computing (STOC), 31–40, 2006.

[Ber00] P. Berman: *A $d/2$ Approximation for Maximum Weight Independent Set in d-Claw Free Graphs.* Proceedings of the 7th Scandinavian Workshop on Algorithms Theory (SWAT). Lecture Notes in Computer Science, Springer, Volume 1851, 31–40, 2000.

[BF94] P. Berman, M. Fürer: *Approximating Maximum Independent Set in Bounded Degree Graphs.* Proceedings of the Fifth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), 365–371, 1994.

[BK03a] P. Berman, M. Karpinski: *Improved Approximation Lower Bound on Small Occurrence Optimization.* Electronic Colloquium on Computational Complexity, Report No. 8, 2003.

[BK03b] P. Berman, P. Krysta: *Optimizing Misdirection.* Proceedings of the 14th Annual Symposium on Discrete Algorithms (SODA), 192–201, 2003.

[BO04] D. Bienstock, N. Ozbay: *Tree-width and the Sherali-Adams Operator.* Discrete Optimization **1**, 13–21, 2004.

[Cal64] M. Całczyńska-Karłowicz: *Theorem on Families of Finite Sets.* Bulletin de l'Académie Polonaise des Sciences. Série des Sciences Mathématiques, Astronomiques et Physiques **12**, 87–89, 1964.

[CH99] B. Chandra, M.M. Halldórsson: *Greedy Local Improvement and Weighted Set Packing Approximation.* Proceedings of the 10th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), 169–176, 1999.

[CMM07] M. Charikar, K. Makarychev, Y. Makarychev: *Local Global Tradeoffs in Metric Embeddings.* Proceedings of the 48th Annual Symposium on Foundations of Computer Science (FOCS), 713–723, 2007.

[CMM09] M. Charikar, K. Makarychev, Y. Makarychev: *Integrality Gaps for Sherali-Adams Relaxations.* Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC), 283–292, 2009.

[CMWY08] F. Chataigner, G. Manic, Y. Wakabayashi, R. Yuster: *Approximation Algorithms and Hardness Results for the Clique Packing Problem*, Discrete Applied Mathematics **157**(7), 1396–1406, 2008.

[Coja05] A. Coja-Oghlan: *The Lovász Number of Random Graphs.* Combinatorics, Probability and Computing **14**(4), 439–465, 2005.

[EL73] P. Erdös, L. Lovász: *Problems and Results on 3-chromatic Hypergraphs and Some Related Questions.* In: Infinite and Finite Sets (A. Hajnal, R Rado, V.T. Sós, Eds.), Proceedings of Colloquia Mathematica Societatis János Bolyai **10**, 609–627. Hungary: Keszthely 1973.

[Fei95] U. Feige: *Randomized Graph Products, Chromatic Numbers, and the Lovász ϑ-Function.* Proceedings of the 27th Annual ACM Symposium on Theory of Computing (STOC), 635–640, 1995.

[FK03] U. Feige, R. Krauthgamer: *The Probable Value of the Lovász–Schrijver Relaxations for Maximum Independent Set.* SIAM Journal on Computing **32**(2), 345–370, 2003.

[Fur81] Z. Füredi: *Maximum Degree and Fractional Matchings in Uniform Hypergraphs.* Combinatorica **1**(2), 155–162, 1981.

[FKS93] Z. Füredi, J. Kahn, P. Seymour: *On the Fractional Matching Polytope of a Hypergraph.* Combinatorica **13**(2), 167–180, 1993.

[GRS03] C.P. Gomes, R.G. Regis, D.B. Shmoys: *An Improved Approximation Algorithm for the Partial Latin Square Problem.* Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), 832–833, 2003.

[GLS81] M. Grötschel, L. Lovász, A. Schrijver: *The Ellipsoid Method and its Consequences in Combinatorial Optimization.* Combinatorics **1**, 169–197, 1981.

[GLS84] M. Grötschel, L. Lovász, A. Schrijver: *Polynomial Algorithms for Perfect Graphs.* Annals of Discrete Mathematics **21**, 325–356, 1984.

[GLS88] M. Grötschel, L. Lovász, A. Schrijver: *Geometric Algorithms and Combinatorial Optimization.* Springer, New York, 1988.

[Hal95] M.H. Halldórsson: *Approximating Discrete Collections via Local Improvments.* Proceedings of the 6th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), 160–169, 1995.

[HR97] M.M. Halldórsson, J. Radhakrishnan: *Greed is Good: Approximating Independent Sets in Sparse and Bounded-degree Graphs.* Algorithmica **18**(1), 145–163, 1997.

[HJKS07] I. Hajirasouliha, H. Jowhari, R. Kumar and R. Sundaram: *On Completing Latin Squares.* Proceedings of the 24th International Symposium on Theoretical Aspects of Computer Science (STACS). Lecture Notes in Computer Science, Springer, Volume 4393, 524–535, 2007.

[HR06] R. Hassin, S. Rubinstein: *An Approximation Algorithm for Maximum Triangle Packing.* Discrete Applied Mathematics **154**(6), 971–979, 2006.

[Has99] J. Håstad: *Clique is Hard to Approximate within $n^{1-\epsilon}$.* Acta Math **182**(1), 105–142, 1999.

[HSS03] E. Hazan, M. Safra, O. Schwartz: *On the Complexity of Approximating k-Dimensional Matching.* Proceedings of APPROX 2003, Lecture Notes in Computer Science, Springer, Volume 2764, 59–70, 2003.

[HSS06] E. Hazan, M. Safra, O. Schwartz: *On the Complexity of Approximating k-Set Packing.* Computational Complexity **15**(1), 20–39, 2006.

[HS89] C.A.J. Hurkens, A. Schrijver: *On the Size of 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 on Discrete Mathematics **2**, 68–72, 1989.

[Jain01] K. Jain: *A Factor 2 Approximation Algorithm for the Generalized Steiner Network Problem.* Combinatorica **21**(1), 39–60, 2001.

[Kann91] V. Kann: *Maximum Bounded 3-Dimensional Matching is MAX SNP-complete.* Information Processing Letters **37**(1), 27–35, 1991.

[Knu94] D.E. Knuth: *The Sandwich Theorem.* The Electronic Journal of Combinatorics **1**, 1–49, 1994.

[Lau03] M. Laurent: *A Comparison of the Sherali-Adams, Lovász-Schrijver, and Lasserre Relaxations for 0-1 Programming.* Mathematics of Operations Research **28**(3), 470–496, 2003.

[Lov79] L. Lovász: *On the Shannon Capacity of a Graph.* IEEE Transactions on Information Theory **25**(1), 1–7, 1979.

[LS91] L. Lovász, A. Schrijver: *Cones of Matrices and Set-Functions and 0-1 Optimization.* SIAM Journal on Optimization **1**(2), 166–190, 1991.

[MM09] A. Magen, M. Moharrami: *Robust Algorithms for Maximum Independent Set on Minor-free Graphs based on the Sherali-Adams Hierarchy.* Proceedings of APPROX 2009, 258–271, 2009.

[MS09] C. Mathieu and A. Sinclair: *Sherali-Adams Relaxations of the Matching Polytope.* Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC), 293–302, 2009.

[MN08] J. Matoušek and J. Nešetřil: *An Invitation to Discrete Mathematics.* Oxford: Clarendon Press, 2008.

[Pad73] M.W. Padberg: *On the Facial Structure of Set Packing Polyhedra.* Mathematical Programming **5**(1), 199–215, 1973.

[Sch08] G. Schoenebeck: *Linear Level Lasserre Lower Bounds for Certain k-CSPs.* Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science (FOCS), 593–602, 2008.

[Tre01] L. Trevisan: *Non-Approximability Results for Optimization Problems on Bounded Degree Instances.* Proceedings of the 33rd Annual ACM Symposium on Theory of Computing (STOC), 453–461, 2001.

[Tul09] M. Tulsiani: *CSP Gaps and Reductions in the Lasserre Hierarchy.* Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC), 303–312, 2009.

[Tuza85] Zs. Tuza: *Critical Hypergraphs and Intersecting Set-pair Systems.* Journal of Combinatorial Theory (B) **39**, 134–145, 1985.

[Tuza87] Zs. Tuza: *On Two Intersecting Set Systems and k-continuous Boolean functions.* Discrete Applied Mathematics **16**, 183–185, 1987.