Independent Sets in Bounded-Degree Hypergraphs [1]

Magnús M. Halldórsson, Elena Losievskaja *

School of Computer Science, Reykjavik University, 103 Reykjavik, Iceland

Department of Computer Science, University of Iceland, 107 Reykjavik, Iceland

Abstract

In this paper we analyze several approaches to the Maximum Independent Set (MIS) problem in hypergraphs with degree bounded by a parameter Δ. Since independent sets in hypergraphs can be strong and weak, we denote by MIS (MSIS) the problem of finding a maximum weak (strong) independent set in hypergraphs, respectively. We propose a general technique that reduces the worst case analysis of certain algorithms on hypergraphs to their analysis on ordinary graphs. This technique allows us to show that the greedy algorithm for MIS that corresponds to the classical greedy set cover algorithm has a performance ratio of $(\Delta + 1)/2$. It also allows us to apply results on local search algorithms on graphs to obtain a $(\Delta + 1)/2$ approximation for the weighted MIS and $(\Delta + 3)/5 - \epsilon$ approximation for the unweighted case. We improve the bound in the weighted case to $\lceil(\Delta + 1)/3\rceil$ using a simple partitioning algorithm. We also consider another natural greedy algorithm for MIS that adds vertices of minimum degree and achieves only a ratio of $\Delta - 1$, significantly worse than on ordinary graphs. For MSIS, we give two variations of the basic greedy algorithm and describe a family of hypergraphs where both algorithms approach the bound of Δ.

Key words: approximation algorithms, maximum independent set, hypergraph

* The corresponding author. Tel.: + 354 869 3041; fax: +354 525 4632.
Email addresses: `mmh@ru.is` (Magnús M. Halldórsson), `elenal@hi.is` (Elena Losievskaja).

[1] Research funded by grants of the Icelandic Research Fund and the Research Fund of the University of Iceland.

1 Introduction

In this paper we consider the independent set problem in hypergraphs. A *hypergraph* H is a pair (V, E), where $V = \{v_1, \ldots, v_n\}$ is a discrete set of vertices and $E = \{e_1, \ldots, e_m\}$ is a collection of subsets of V, or (hyper)edges. A hypergraph is *simple* if no edge is a subset of another edge. An *independent* set in H is a subset of V that doesn't contain any edge of H, also referred to as a *weak* independent set [2]. If an independent set in H intersects any edge in E in at most one element, then it is said to be a *strong* independent set [2]. Let MIS (MSIS) denote the problem of finding a maximum unweighted weak (strong) independent set in hypergraphs, respectively. If we consider a weighted version of MIS (MSIS), we state it explicitly.

MIS is of fundamental interest, both in practical and theoretical aspects. It arises in various applications in data mining, image processing, database design, parallel computing and many others. MIS is intimately related with classical covering problems. The vertices not contained in a weak independent set form a vertex cover, or a *hitting set*. Moreover, a set cover in the dual of a hypergraph (replacing each set by a vertex and including a set for the incidences of each original node) is equivalent to a hitting set in the original hypergraph. Thus, in terms of optimization, MIS is equivalent to the Hitting Set and the Set Cover problems.

Numerous results are known about independent sets in hypergraphs, including approximation algorithms for MIS in [17] and [20]. The focus of the current work is on bounded-degree hypergraphs, where each vertex is of degree at most Δ. Given that both MIS and MSIS generalize the independent set problem in graphs, the problem is NP-hard to approximate within a factor $\Delta/2^{O\left(\sqrt{\log \Delta}\right)}$ unless $P = NP$ [25].

In the case of graphs (2-uniform hypergraphs), there is no distinction between weak and strong independent sets. Thus, we denote by MIS the problem of finding a maximum independent set in graphs. Various approximation algorithms have been given for MIS in graphs. Halldórsson and Radhakrishnan [14] showed that the minimum-degree greedy algorithm approximates unweighted MIS within a factor of $\frac{\Delta+2}{3}$. A simple partitioning algorithm due to Halldórsson and Lau [13] gives a $(\Delta + 2)/3$-approximation of weighted MIS. A better approximation ratio for unweighted MIS is $(\Delta + 3)/5$ obtained by Berman and Fujito [4] using a local search algorithm. For large values of Δ, the best approximation is obtained by using semi-definite programming, with a ratio of $O(\Delta \log\log \Delta / \log \Delta)$ due to Vishwanathan [26] (and also in the weighted case, shown independently by Halldórsson [12] and Halperin [15]).

The MSIS problem can be turned into an independent set problem in graphs, by replacing each hyperedge with a clique (assuming that a hypergraph has no singletons, otherwise we can always delete such vertices from the hypergraph, because they can not belong to any independent set). The reason for considering the problem as a hypergraph problem is that the degrees in the hypergraph can be much smaller than in the corresponding clique graph. If the hypergraph is of degree Δ, then the corresponding clique graph contains no $\Delta+1$-claw, where a k-claw is an induced star on k edges. The work of Hurkens and Schrijver [18] established that a natural local improvement method attains a performance ratio of $k/2+\epsilon$, for any fixed $\epsilon > 0$, on $k+1$-claw free graphs. Another local search algorithm by Berman [3] approximates weighted MIS in $(d+1)$-claw free graphs within a factor of $(d+1)/2$, which implies also a $\Delta/2$-approximation. A strong hardness result of $\Omega(\frac{\Delta}{\ln \Delta})$ is known for MSIS, due to Hazan, Safra and Schwartz [16]. The focus of our study of MSIS is to consider natural greedy methods and establish tight bounds on their performance ratio.

One of the most extensively studied heuristics of all times is the *greedy set cover* algorithm, which repeatedly adds to the cover the set with the largest number of uncovered elements. In spite of its simplicity, it is in various ways also one of the most effective. Johnson [19] and Lovász [22] showed that it approximates the Set Cover problem within $H_n \leq \ln +1$ factor, which was shown by Feige [10] to be the best possible up to a lower order term. Generalizations to weights [8] and submodular functions [27] also yield equivalent ratios. And under numerous variations on the objective function does it still achieve the best known/possible performance ratio, e.g. Sum Set Cover [11] and Entropy Set Cover [6]. Bazgan, Monnot, Paschos and Serrière [1] studied the *differential approximation ratio* of the greedy set cover algorithm, this ratio measures how many sets are *not* included in the cover. When viewed on the dual hypergraph, this is equivalent to studying the performance ratio of the greedy set cover algorithm for MIS. They proved that when modified with a post-processing phase, it has a performance ratio of at most $\Delta/1.365$ and at least $(\Delta+1)/4$. Caro and Tuza [7] showed that the greedy set cover algorithm applied to MIS in r-uniform hypergraphs always finds a weak independent set of size at least $\Theta\left(n/\Delta^{\frac{1}{r-1}}\right)$. Thiele [24] extended their result to non-uniform hypergraphs and gave a lower bound on the size of an independent set found by a greedy algorithm as a complicated function of the number of edges of different sizes incident on each vertex in a hypergraph.

Another popular algorithm design technique is local search. This technique is based on the concept of a *neighborhood* - a set of solutions close to a given solution S. The idea is to start with some (arbitrary) solution S and iteratively replace S by a better solution found in the neighborhood of S. Local search gives the best approximations of weighted and unweighted MIS in bounded-degree graphs for small values of Δ, due to Berman [3] and Berman and Fujito [4]. Bazgan, Monnot, Paschos and Serrière [1] considered a simple 2-OPT local search algorithm to approximate MIS in hypergraphs and proved a tight bound of $(\Delta + 1)/2$.

Another simple approach in approximation algorithm design is partitioning. The strategy is to break the problem into a set of easier subproblems, solve each subproblem and output the largest of the found solutions. This approach yields $O(n \log\log n / \log n)$ and $\lceil(\Delta + 1)/3\rceil$ approximations to the weighted MIS in graphs, as shown in [12]. In spite of its simplicity, partitioning has not been used before to approximate MIS in hypergraphs.

In this paper we analyze greedy, local search and partitioning approaches to approximate weighted and unweighted MIS and MSIS in bounded-degree hypergraphs. We describe a general technique that reduces the worst case analysis of certain algorithms to their analysis on ordinary graphs. Given an approximation algorithm A, this technique, called *shrinkage reduction*, truncates a hypergraph H to a graph G such that an optimal solution on H is also an optimal solution in G, and A produces the same worst approximate solution on H and G. This technique can be applied to a wide class of algorithms and problems on hypergraphs. For example, this technique allows us to show that the greedy algorithm for MIS that corresponds to the classical greedy set cover algorithm has a performance ratio of $(\Delta + 1)/2$, improving the bounds obtained by Bazgan et al. [1]. In addition, while their analysis required a post-processing phase, our bound applies to the greedy algorithm alone. It also allows us to apply results on local search algorithms on graphs to obtain a $(\Delta + 1)/2$ approximation for weighted MIS and $(\Delta + 3)/5 + \epsilon$ approximation for unweighted MIS. We improve the bound in the weighted case to $\lceil(\Delta + 1)/3\rceil$ using a simple partitioning algorithm. Finally, we show that another natural greedy algorithm for MIS, that adds vertices of minimum degree, achieves only a ratio of $\Delta - 1$, significantly worse than on ordinary graphs.

For MSIS we describe two greedy algorithms: one constructs an independent set by selecting vertices of minimum degree, the other selects vertices with the fewest neighbors. We show that both algorithms have a performance ratio of Δ, and this bound is tight. However, in r-uniform hypergraphs the performance ratio of all greedy algorithms is improved: for MIS to $\Theta\left(\Delta^{\frac{1}{r-1}}\right)$ and $1 + \frac{\Delta-1}{r}$, respectively; for MSIS to $\Delta - \frac{\Delta-1}{r}$ for both greedy algorithms.

The paper is organized as follows. After giving essential definitions of various hypergraph properties, we present the shrinkage reduction technique and apply it to the analysis of local search and greedy algorithms to MIS in Section 3. We conclude Section 3 with the application of simple partitioning and the minimum-degree greedy algorithms to MIS. In Section 4 we describe two greedy algorithms for MSIS.

2 Definitions

Given a hypergraph $H = (V, E)$, let n and m be the number of vertices and edges in H. The *degree* of a vertex v is the number of edges incident on v. We denote by Δ and $\overline{d}$ the maximum and the average degree in the hypergraph, respectively. In a *bounded-degree* hypergraph Δ is a constant. A hypergraph is Δ-*regular* if all vertices have the same degree Δ.

The *rank* r of a hypergraph H is the maximum edge size in H. A hypergraph is r-*uniform* if all edges have the same cardinality r. By a t-*edge* we mean an edge of size t.

A vertex u is a *neighbor* of a vertex v, if there exist an edge $e \in E$ that includes both u and v. Given a vertex $v \in V$, we denote by $N(v)$ a set of neighbors of v. Let $N_k(v) = \{u \in V : \exists e \in E, (u, v) \in e, |e| = k\}$ be a set of neighbors of v in edges of size k. Given a set $U \subseteq V$, let $N(U) = \{v \in V \backslash U : \exists u \in U, \exists e \in E, (u, v) \in e\}$ a set of neighbors of vertices in U.

A *hyperclique* is a hypergraph in which each vertex is a neighbor of all other vertices. Note, that a hyperclique need not be a uniform hypergraph. By analogy with a graph being a 2-uniform hypergraph, a clique is a 2-uniform hyperclique.

A *(hyper)path* in a hypergraph is a sequence of edges $e_1, e_2, \ldots, e_p$ such that $e_i \cap e_{i+1} \neq \emptyset$ for any $1 \leq i \leq p-1$ and $e_i \cap e_j = \emptyset$ for any i, j such that $|i - j| > 1$.

An n-*star* is a tree on $n + 1$ nodes with one node of degree n (the *root* of the star) and the others of degree 1 (the *endpoints* of the star).

We say that a hypergraph $H'(V', E')$ is *induced* in $H(V, E)$ on the vertex set $V' \subset V$, if $E' = \{e \in E | e \subseteq V'\}$. By *deleting* a vertex v from a hypergraph H we mean just one operation: $V = V \backslash \{v\}$, and by *deleting v with all incident edges* we mean two operations: $V = V \backslash \{v\}$ and $E = E \backslash \{e \in E | v \in e\}$.

In the remainder, we let $\mathcal{H}$ and $\mathcal{G}$ be the collections of all hypergraphs and graphs, respectively. We denote by H a hypergraph in $\mathcal{H}$ and by G a graph in $\mathcal{G}$, respectively. By a *cover* we mean a hitting set in H or a vertex cover in G.

3 Weak Independent Set

We describe three different approaches to weighted and unweighted MIS in bounded-degree hypergraphs: local search, greedy and partitioning. We also present a general reduction technique for the worst case analysis of approximation algorithms on hypergraphs and apply it to local search and greedy algorithms.

3.1 Shrinkage Reduction

Shrinkage reduction is a general technique that reduces the worst case analysis of algorithms on hypergraphs to their analysis on graphs. It is based on a *shrinkage* hypergraph, or *shrinkage* for short.

Definition 3.1 *A hypergraph H' is a* shrinkage *of H if $V(H') = V(H)$, $|E(H')| = |E(H)|$ and for any edge $e \in E(H)$ there exist an edge $e' \in E(H')$ such that $e' \subseteq e$. In other words, the edges of H might be truncated in H' into sets of smaller size (and at least 2).*

Shrinkage reduction works for *hereditary* optimization problems. Given an instance I, an optimization problem consists of a set of feasible solutions $\mathcal{S}_I$ and a function $w : \mathcal{S}_I \to R^+$ assigning a non-negative cost to each solution $S \in \mathcal{S}_I$.

Definition 3.2 *An optimization problem on hypergraphs is* hereditary*, if for any shrinkage H' of a hypergraph H it satisfies $\mathcal{S}_{H'} \subseteq \mathcal{S}_H$.*

Many problems on hypergraphs are hereditary, including the Minimum Hitting Set, the Maximum Independent Set, the Minimum Coloring and the Shortest HyperPath. An example of non-hereditary problem is the Longest HyperPath. Given a hereditary problem, the essence of shrinkage reduction is the following.

Proposition 3.3 *Let A be an approximation algorithm for a hereditary problem. Suppose we can construct a shrinkage graph G of a hypergraph H such that an optimal solution in H is also an optimal solution in G and A produces the same worst approximate solution on H and G, then the performance ratio of A on hypergraphs is no worse than on graphs.*

Note, that Proposition 3.3 applies also to non-deterministic (and randomized) approximation algorithms.

It is not easy to give a general rule on how to construct a shrinkage for an arbitrary approximation algorithm. In the following sections we describe reductions for the greedy set cover and local search algorithms for weighted and unweighted MIS in bounded-degree hypergraphs. The comparison of the $GreedyMAX$ and the $GreedyMIN$ algorithms, described in Sections 3.3 and 3.5 respectively, suggests that the shrinkage reduction technique might be applicable only to algorithms that don't alter edge sizes during the execution.

The idea of the local search approach is to start with a (arbitrary) solution and continually replace it by a better solution found in its neighborhood while possible. We need formal definitions to determine what a "better solution" and a "neighborhood" mean.

A *neighborhood function* Γ maps a solution $S \in \mathcal{S}_I$ into a set of solutions $\Gamma_I(S) \subseteq \mathcal{S}_I$, called the *neighborhood* of S. A feasible solution $\tilde{S}$ is *locally optimal w.r.t.* Γ, or Γ*-optimal* for short, if it satisfies $w(\tilde{S}) \leq w(S)$ $(w(\tilde{S}) \geq w(S))$ for all $S \in \Gamma_I(S)$ for a minimization (maximization) problem. A feasible solution S^* is *globally optimal*, or *optimal* for short, if it satisfies $w(S^*) \leq w(S)$ $(w(S^*) \geq w(S))$ for all $S \in \mathcal{S}_I$ for a minimization (maximization) problem. To specify more precisely the neighborhood functions used in our local search algorithms, we need the following definition.

Definition 3.4 *A neighborhood function* Γ *is said to be* edge-monotone *for a hereditary problem on hypergraphs if for any shrinkage* H' *of a given hypergraph* H *and any solution* $S \in \mathcal{S}_{H'}$ *the neighborhood of* S *satisfies* $\Gamma_{H'}(S) \subseteq \Gamma_H(S)$.

In other words, edge-monotonicity means that edge reduction can only decrease the neighborhood size.

A Γ-optimal algorithm is a local search algorithm that given an instance I, starts with a (arbitrary) solution S and repeatedly replaces it by a better solution found in $\Gamma_I(S)$ until S is Γ-optimal. The approximation ratio $\varrho_{\Gamma,I}$ of a Γ-optimal algorithm on a instance I is the maximum ratio between the weights of Γ-optimal and optimal solutions over all Γ-optimal solutions on I, i.e. $\varrho_{\Gamma,I} = \max\limits_{\forall \tilde{S} \in \mathcal{S}_I} \frac{w(\tilde{S})}{w(S^*)}$ $\left(\varrho_{\Gamma,I} = \max\limits_{\forall \tilde{S} \in \mathcal{S}_I} \frac{w(S^*)}{w(\tilde{S})}\right)$ for a minimization (maximization) problem. The performance ratio $\rho_{\Gamma,\mathcal{I}}$ of a Γ-optimal algorithm is the worst approximation ratio over all instances I in the class of instances $\mathcal{I}$.

In the following theorem we show that if a neighborhood function Γ is edge-monotone, then for the Minimum Cover problem the analysis of a Γ-optimal algorithm on hypergraph reduces to the analysis of this algorithm on graphs. The reduction is based on the construction of a shrinkage graph with special properties. Note, that a shrinkage graph is needed only for the analysis, but not for the Γ-optimal algorithm itself.

Theorem 3.5 *Given an edge-monotone neighborhood function* Γ *and a hypergraph* H *with an optimal cover* S^* *and a* Γ*-optimal cover* $\tilde{S}$*, there exists a shrinkage graph* G *of* H *on which* S^* *and* $\tilde{S}$ *are also optimal and* Γ*-optimal covers, respectively.*

Proof: Given H, S^* and $\tilde{S}$, we construct a shrinkage G as follows. From each edge e in $E(H)$, we arbitrarily pick vertices u and v such that $\{u, v\} \cap \tilde{S} \neq \emptyset$ and $\{u, v\} \cap S^* \neq \emptyset$, and add (u, v) to $E(G)$.

Any edge in $E(G)$ contains at least one vertex from $\tilde{S}$ and at least one vertex from S^*, and so $\tilde{S}$ and S^* are covers in G, i.e. $\tilde{S}, S^* \in \mathcal{S}_G$. Since G is a shrinkage of H and the Minimum Cover problem is hereditary, $\mathcal{S}_G \subseteq \mathcal{S}_H$ by definition. For all $S \in \mathcal{S}_H$ we have $w(S^*) \leq w(S)$, and so $w(S^*) \leq w(S)$ for all $S \in \mathcal{S}_G$. Thus, S^* is an optimal cover in G. The local optimality of $\tilde{S}$ in G follows by the same argument and the fact that Γ is edge-monotone. □

Corollary 3.6 *If a neighborhood function Γ is edge-monotone for MIS, then $\rho_{\Gamma,\mathcal{H}} \leq \rho_{\Gamma,\mathcal{G}}$.*

Proof: Given a hypergraph $H(V, E)$, the vertices not contained in a weak independent set I form a vertex cover S in H, i.e. $I = V \backslash S$. Given an edge-monotone neighborhood function Γ for MIS, we define a new neighborhood function $\Gamma'(S) = \{S' : V \backslash S' \in \Gamma(V \backslash S)\}$. Note, that $\Gamma'(S)$ is edge-monotone for the Hitting Set problem. Moreover, if I^* and $\tilde{I}$ are optimal and Γ-optimal weak independent sets in H, then $S^* = V \backslash \tilde{I}^*$ and $\tilde{S} = V \backslash \tilde{I}$ are optimal and Γ-optimal covers in H, respectively. The claim then follows from Theorem 3.5. □

The simplest local search algorithm for MIS is t-Opt, which repeatedly tries to extend the current solution by deleting t elements while adding $t+1$ elements. It is easy to verify that the corresponding neighborhood function $\Gamma(S) = \{S' \in \mathcal{S}_H : |S \oplus S'| \leq t\}$ defined on $\mathcal{S}_H$ is edge-monotone (where $\oplus$ is the symmetric difference). Then, the following two theorems are straightforward from Corollary 3.6 and the results of Hurkens and Schrijver on graphs [18].

Theorem 3.7 *t-Opt approximates MIS within $\Delta/2 + \epsilon$, where $\lim_{t\to\infty} \epsilon(t) = 0$.*

Theorem 3.8 2*-Opt approximates MIS within* $(\Delta + 1)/2$.

Theorem 3.9 *For every $\epsilon > 0$, MIS can be approximated within $(\Delta+3)/5+\epsilon$ for even Δ and within $(\Delta + 3.25)/5 + \epsilon$ for odd Δ.*

Proof: We extend the algorithm $SIC_{\Delta,k}$ of Berman and Fürer [5] for MIS in bounded degree graphs to the hypergraph case. Given a hypergraph $H(V, E)$ and a weak independent set A in H, let B_A equal $V - A$ if the maximum degree of H is three, and otherwise equal the set of vertices that have at least two incident edges with vertices in A. Let $Comp(A)$ be the subhypergraph induced by B_A. The formal description of the algorithm is given in Figure 3.2.

ALGORITHM $HSIC\ (H,\ \Delta,\ k)$

If $\Delta\ \leq\ 2$ then compute MIS exactly and stop
Let A be any maximal weak independent set
Repeat
  Do all possible local improvements of size $O(k\ \log\ n)$
  If $\Delta\ = 3$ then $l\ =\ 1$ else $l = 2$
  Recursively apply $HSIC(Comp(A),\ \Delta - l,\ k)$
    and select the resulting weak independent set if it is bigger
Until A has no improvements

Fig. 1. The algorithm $HSIC$

There are two neighborhood functions in $HSIC$. The first function which maps a solution A to a set of all possible local improvements of size $O(k \log n)$, is t-optimal with $t = O(k \log n)$, and thereofre edge-monotone. The second function, which maps a solution A to a set of weak independent sets in $Comp_H(A)$, is edge-monotone, because shrinking H to H' reduces the degree of some vertices, implying $B_A(H') \subseteq B_A(H)$. Consequently, a weak independent set in $Comp_{H'}(A)$ is also a weak independent set in $Comp_H(A)$. Thus, both neighborhood functions are edge-monotone and the performance ratio of $HSIC$ is no worse than the performance ratio of $SIC_{\Delta,k}$ by Corollary 3.6. □

Theorem 3.10 *Weighted MIS can be approximated within* $(\Delta + 1)/2$ *on hypergraphs of a constant rank* r*.*

Proof: We extend the algorithm $SquareIMP$ of Berman [3] for weighted MIS in bounded degree graphs to the hypergraph case. Let S be a weak independent set in H. We say that (A, B) is an *improvement* of S, if there is a vertex $v \in S$ such that $A \subseteq N(v) \cap (V \backslash S)$, $B \subseteq N(A) \cap S$, $(S \backslash B) \cup A$ is a weak independent set and $w^2((S \backslash B) \cup A) > w^2(S)$. The formal description of the algorithm in Figure 3.2.

ALGORITHM $HSquareIMP\ (H)$

$S\ \leftarrow\ \emptyset$
While there exist an improvement (A, B) of S
  $S\ \leftarrow\ (S \setminus B)\ \cup\ A$
Output S

Fig. 2. The algorithm $HSquareIMP$

The neighborhood function in $HSquareIMP$ is edge-monotone. Shrinking H to H' reduces the degree of some vertices and so every improvement A, B of S in H' is also an improvement of S in H. Hence, the performance ratio of $HSquareIMP$ is no worse than the performance ratio of $SquareIMP$ by Corollary 3.6.

Note, that finding an improvement (A, B) takes $O(n2^{\Delta^2(r-2)(r-1)})$ steps. Namely, in the worst case we check every vertex $v \in S$, every possible subset $A \subseteq N(v) \cap (V \backslash S)$ and every possible subset $B \subseteq N(A) \cap S$ to see whether $(S \backslash B) \cup A$ is a weak independent set and $w^2((S \backslash B) \cup A) > w^2(S)$. Since $|N(v) \cap (V \backslash S)| \leq \Delta(r-2)$, there are at most $2^{\Delta(r-2)}$ possible A-sets. Similarly, since $|N(A) \cap S| \leq \Delta(r-2)(\Delta(r-1)-1)$, there are at most $2^{\Delta(r-2)(\Delta(r-1)-1)}$ possible B-sets. In total, we consider at most $2^{\Delta^2(r-2)(r-1)}$ possible pairs (A, B) for every vertex $v \in S$ until an improvement is found. $\square$

3.3 The GreedyMAX Algorithm

The idea of the greedy approach is to construct a solution by repeatedly selecting the best candidate on each iteration. There are two variations, called *GreedyMAX* and *GreedyMIN*, depending on whether we greedily reject or add vertices.

The *GreedyMAX* algorithm constructs a cover S by adding a vertex of maximum degree, deleting it with all incident edges from the hypergraph, and iterating until the edge set is empty. It then outputs the remaining vertices as a weak independent set I. The formal description of the algorithm is given in Figure 3.3.

```
ALGORITHM GreedyMAX (H)
    S = ∅
    While the edge set is not empty
      Add a vertex v of maximum degree to S
      Delete v with all incident edges on v from H
    Output I = V \ S
```

Fig. 3. The algorithm *GreedyMAX*

Given a hypergraph $H(V, E)$, let S^* be a minimum cover. Then, the performance ratio of *GreedyMAX* is:

$$\rho = \max_{\forall H} \frac{n - |S^*|}{n - |S|} \ . \tag{1}$$

The analysis has two parts. First we prove that the worst case for *GreedyMAX* occurs on graphs. Namely, we describe how to reduce any hypergraph to a graph (actually, a multigraph) G for which *GreedyMAX* has no better performance ratio. We then show that the bound actually holds for (multi)graphs.

Lemma 3.11 *Given a hypergraph H with a minimum cover S^*, there exists* *a shrinkage G of H on which S^* is still a cover and where GreedyMAX constructs the same cover for G as for H.*

Proof: The proof is by induction on s, the number of iterations of *GreedyMAX*. For the base case, $s = 0$, the claim clearly holds for the unchanged empty graph.

Suppose now that the claim holds for all hypergraphs for which *GreedyMAX* selects $s - 1 \geq 0$ vertices. Let u_1 be the first vertex chosen by *GreedyMAX*, $E(u_1)$ be the set of incident edges, and H_1 be the remaining hypergraph after deleting u_1 with all incident edges. Based on $E(u_1)$, we form a set $E'(u_1)$ of ordinary edges as follows. If u_1 is contained in both S and S^*, then for each edge e in $E(u_1)$ we pick an arbitrary vertex v from e and add (u_1, v) to $E'(u_1)$. If u_1 is only in S and not in S^*, then for each edge e in $E(u_1)$ we pick an arbitrary vertex u from e that is contained in S^* and add (u_1, u) to $E'(u_1)$; such a vertex u must exist, since e is covered by S^*. This completes the construction of $E'(u_1)$.

By the inductive hypothesis, there is a shrinkage G_1 of H_1 with a greedy cover of $S \setminus \{u_1\}$ and G_1 is still covered by S^*. We now form the multigraph G on the same vertex set as H with the edge set $E'(u_1) \cup E(G_1)$, and claim that it satisfies the statement of the lemma. Since G_1 is covered by S^* and all edges of $E'(u_1)$ are also covered by vertices of S^*, S^* covers all edges of G. The edge shrinkage only decreases the degrees of vertices, but does not affect the degree of u_1. Therefore, u_1 remains the first vertex chosen by *GreedyMAX* and, by induction, the vertices chosen from G_1 are the same as those chosen from H_1. Hence, *GreedyMAX* outputs the same solution on G as on H, completing the lemma. □

From Lemma 3.11 it follows immediately that the performance ratio of *GreedyMAX* on hypergraphs is no worse than on graphs. Sakai, Togasaki, and Yamazaki [23] obtained a lower bound on the size of weighted independent set I produced by a weighted generalization of *GreedyMAX* on graphs. In unweighted case this bound reduces to a Caro-Wei improvement of the Turan bound on graphs $|I| \geq \sum_{v \in V} \frac{1}{d(v)+1}$. For completeness we give below the proof from [23] adapted for unweighted multigraphs.

Lemma 3.12 *Given a (multi)graph $G = (V, E)$, GreedyMAX finds an independent set of size at least $\sum_{v \in V} \frac{1}{d(v)+1}$.*

Proof: Let s be the number of iterations of *GreedyMAX* on G. For $0 \leq i \leq s$, let G_i be the remaining (multi)graph after i iterations. We denote by $d_{G_i}(v)$ and $N_{G_i}(v)$ the degree and the neighborhood of a vertex $v \in V(G_i)$. Note, that since G_i is a multigraph, $N_{G_i}(v)$ is a multiset and $d_{G_i}(v) = |N_{G_i}(v)|$. For a vertex $u \in N_{G_i}(v)$ let $e_{G_i}(v,u)$ be the number of multiple edges (v,u) in G_i. Let $f(G_i) = \sum_{u \in V(G_i)} \frac{1}{d_{G_i}(u)+1}$ be a potential function on a graph G_i. We show that $f(G_{i+1}) \geq f(G_i)$ for $0 \leq i \leq s$. Consequently, $f(G_s) \geq f(G_0)$, where G_0 is the original graph G and G_s is a collection of isolated vertices. Then, $GreedyMAX$ outputs a weak independent set of size at least:

$$|I| = f(G_s) \geq f(G) = \sum_{u \in V(G)} \frac{1}{d_G(u)+1} \ . \tag{2}$$

Let v_i be the vertex chosen by $GreedyMAX$ on the iteration i. Then,

$$\begin{aligned} f(G_{i+1}) &= \sum_{u \in V(G_{i+1})} \frac{1}{d_{G_{i+1}}(u)+1} \\ &= \sum_{u \in V(G_i)} \frac{1}{d_{G_i}(u)+1} - \frac{1}{d_{G_i}(v_i)+1} + \sum_{u \in V(G_i) \cap N_{G_i}(v_i)} \left(\frac{1}{d_{G_{i+1}}(u)+1} - \frac{1}{d_{G_i}(u)+1} \right) \\ &= f(G_i) - \frac{1}{d_{G_i}(v_i)+1} + Y \ , \end{aligned} \tag{3}$$

where

$$\begin{aligned} Y &= \sum_{u \in V(G_i) \cap N_{G_i}(v_i)} \left(\frac{1}{d_{G_{i+1}}(u)+1} - \frac{1}{d_{G_i}(u)+1} \right) \\ &= \sum_{u \in N_{G_i}(v_i)} \frac{1}{e_{G_i}(v,u)} \left(\frac{1}{d_{G_i}(u) - e_{G_i}(v,u)+1} - \frac{1}{d_{G_i}(u)+1} \right) && (4) \\ &\geq |N_{G_i}(v_i)| \min_{u \in N_{G_i}(v_i)} \frac{1}{(d_{G_i}(u) - e_{G_i}(v,u)+1)\,(d_{G_i}(u)+1)} \\ &\geq |N_{G_i}(v_i)| \min_{u \in N_{G_i}(v_i)} \frac{1}{d_{G_i}(u)\,(d_{G_i}(u)+1)} \\ &\geq \frac{|N_{G_i}(v_i)|}{d_{G_i}(v_i)\,(d_{G_i}(v_i)+1)} && (5) \\ &= \frac{1}{d_{G_i}(v_i)+1} && (6) \end{aligned}$$

and (4) follows from $d_{G_{i+1}}(u) = d_{G_i}(u) - e_{G_i}(v,u)$, which is minimized when $e_{G_i}(v,u) = 1$; (5) holds by the greedy rule $d_{G_i}(v_i) \geq \max_{u \in G_i} d_{G_i}(u)$. It follows from (3) and (6) that $f(G_{i+1}) \geq f(G_i)$ completing the proof. □

Lemma 3.13 *The performance ratio of GreedyMAX on (multi)graphs is at most* $\frac{\Delta+1}{2}$.

Proof: We show that *GreedyMAX* attains its worst performance ratio on regular graphs. First we refine $\bar{d}$ as follows: let $k \in [0,1]$ be the value so that kn vertices are of degree Δ and the remaining $(1-k)n$ vertices have average degree $d' \leq \Delta - 1$. Then,

$$\bar{d} = k\Delta + (1-k)d' \ . \tag{7}$$

Since each vertex can cover at most Δ of the m edges of the graph, any optimal cover S^* is of size at least

$$|S^*| \geq \frac{m}{\Delta} = \frac{\bar{d}n}{2\Delta} = \frac{n\left(k\Delta + (1-k)d'\right)}{2\Delta} \ . \tag{8}$$

We also rewrite (2) using (7) as

$$|I| \geq \sum_{v \in V} \frac{1}{d(v)+1} \geq \frac{kn}{\Delta+1} + \sum_{v \in V \ : \ d(v) < \Delta} \frac{1}{d(v)+1} \ . \tag{9}$$

Since $f(d) = \frac{1}{d+1}$ is a convex function, we can apply Jensen's inequality[2] to (9):

$$|I| \geq \frac{kn}{\Delta+1} + \frac{(1-k)n}{d'+1} \ . \tag{10}$$

Note, that the same result follows from the harmonic-arithmetic mean inequality applied to (9). Combining (1), (8) and (10) we obtain an upper bound on the performance ratio of *GreedyMAX*:

$$\begin{aligned} \rho &= \max_{\forall H} \frac{n-|S^*|}{n-|S|} = \max_{\forall H} \frac{n-|S^*|}{|I|} \leq \frac{2\Delta - k\Delta - (1-k)d'}{2\Delta\left(\frac{k}{\Delta+1} + \frac{1-k}{d'+1}\right)} \\ &= \frac{(\Delta+1)(d'+1)}{2\Delta}\left(1 + \frac{\Delta - d' - 1}{\Delta + 1 - k(\Delta - d')}\right), \end{aligned} \tag{11}$$

where (11) is maximized when $k=1$, yielding a bound of $\frac{\Delta+1}{2}$. □

Theorem 3.14 *The performance ratio of GreedyMAX on hypergraphs is* $\frac{\Delta+1}{2}$.

Proof:

[2] Jensen's inequality for a convex function f: $\sum_{i=1}^{n} f(x_i) \geq nf\left(\frac{1}{n}\sum_{i=1}^{n} x_i\right)$

The upper bound is straightforward from Lemmas 3.11 and 3.13, because G and H have the same number of edges and the same maximum degree. The edge reduction in $E(H)$ might create multiple edges in $E(G)$, but they don't affect the performance ratio of *GreedyMAX*.

For the lower bound, consider the graph $G_{\Delta+1,\Delta+1}$, formed by a complete bipartite graph missing a single perfect matching. *GreedyMAX* may remove vertices alternately from each side, until two vertices remains as a maximal weak independent set. The optimal solution consists of one of the bipartitions, of size $\Delta+1$. By taking independent copies, this can be extended for arbitrarily large instances. □

Theorem 3.15 *The performance ratio of GreedyMAX in r-uniform hypergraphs is at most* $\left(\frac{r-1}{r}\right)\prod_{i=1}^{\Delta}(1+\frac{1}{i(r-1)}) = \Theta\left(\Delta^{\frac{1}{r-1}}\right)$.

Proof: We assume that $r \geq 3$ because 2-uniform hypergraphs are ordinary graphs and the analysis of the greedy algorithm on graphs is given in Lemma 3.13.

Caro and Tuza [7] showed that *GreedyMAX* always finds an independent set I of size at least:

$$|I| \geq \sum_{v\in V}\prod_{i=1}^{d(v)}\left(1-\frac{1}{i(r-1)+1}\right) = \sum_{v\in V}\prod_{i=1}^{d(v)}\frac{i}{i+\frac{1}{r-1}} = \sum_{v\in V}\frac{d(v)!}{\left(d(v)+\frac{1}{r-1}\right)^{\underline{d(v)}}} \quad ,(12)$$

where $x^{\underline{y}} = x(x-1)\dots(x-y+1)$. The function $f(d) = \frac{d!}{(d+\frac{1}{r-1})^{\underline{d}}} = \binom{d+\frac{1}{r-1}}{d}^{-1}$ is convex, because its first derivative is monotonically increasing on the interval $[1,\Delta]$. Therefore, we can apply Jensen's inequality to (12):

$$|I| \geq n\binom{\bar{d}+\frac{1}{r-1}}{\bar{d}}^{-1}.$$

Any maximum independent set in a r-uniform hypergraph on n vertices contains $n-|S|$ vertices, where S is a minimum hitting set. Since there are at most $\bar{d}n/r$ edges in a r-uniform hypergraph and each vertex from S covers at most Δ edges, there are at least $\frac{\bar{d}n}{r\Delta}$ vertices in S. Then, the performance ratio of *GreedyMAX* is at most

$$\rho \leq \frac{n-\frac{\bar{d}n}{r\Delta}}{n\binom{\bar{d}+\frac{1}{r-1}}{\bar{d}}^{-1}} = \left(1-\frac{\bar{d}}{r\Delta}\right)\binom{\bar{d}+\frac{1}{r-1}}{\bar{d}} \leq \left(1-\frac{1}{r}\right)\binom{\Delta+\frac{1}{r-1}}{\Delta}$$

because $f\left(\bar{d}\right) = \left(1-\frac{\bar{d}}{r\Delta}\right)\binom{\bar{d}+\frac{1}{r-1}}{\bar{d}}$ is maximized when $\bar{d}=\Delta$. □

Theorem 3.16 *The performance ratio of GreedyMAX in r-uniform hypergraphs is at least* $\left(\frac{r-1}{r}\right)\prod_{i=1}^{\Delta}\left(1+\frac{1}{i(r-1)}\right)=\Theta\left(\Delta^{\frac{1}{r-1}}\right)$.

Proof: Let n be a multiple of $\prod_{j=1}^{\Delta}(j(r-1)+1)$ and for any $i \in [1,\Delta]$ let $x_i = \frac{n}{i(r-1)}\prod_{j=i}^{\Delta}\frac{j(r-1)}{j(r-1)+1}$. We define a chain of regular r-uniform hypergraphs $H^{(1)} \subset H^{(2)} \ldots \subset H^{(\Delta-1)} \subset H^{(\Delta)}$, where our hypergraph $H(V,E) = H^{(\Delta)}$.

The first hypergraph $H^{(1)}$ is defined on rx_i vertices and consists of x_1 disjoint edges, i.e $V^{(1)} = \{v_1^{(1)}, \cdots, v_{rx_1}^{(1)}\}$ and $E^{(1)} = \{e_1^{(1)}, \cdots e_{x_1}^{(1)}\}$, where $e_j^{(1)} = \{v_{(j-1)r+1}^{(1)} \cdots v_{jr}^{(1)}\}$ for any $j \in [1, x_1]$. Let $T^{(1)} = E^{(1)}$ and $U^1 = \{v_r^{(1)}, v_{2r}^{(1)}, \cdots, v_{rx_1}^{(1)}\}$. It is easy to see that $H^{(1)}$ is a 1-regular r-uniform hypergraph.

For $i \in [2,\Delta]$, let $y_i = ix_i$. The hypergraph $H^{(i)}$ consists of $H^{(i-1)}$, an additional set of vertices $U^{(i)} = \{u_1^{(i)}, \ldots, u_{x_i}^{(i)}\}$ and an additional set of edges $T^{(i)} = \{t_1^{(i)}, \ldots, t_{y_i}^{(i)}\}$, connecting $U^{(i)}$ to $H^{(i-1)}$, i.e $V^{(i)} = V^{(i-1)} \cup U^{(i)}$ and $E^{(i)} = E^{(i-1)} \cup T^{(i)}$. The first y_{i-1} edges in $T^{(i)}$ are the copies of the edges in $T^{(i-1)}$ with the last vertex in each copy replaced by a vertex from $U^{(i)}$, i.e $t_j^{(i)} = t_j^{(i-1)} \backslash \{v_{jr}^{(i-1)}\} \cup \{u_{\lceil j/i \rceil}^{(i)}\}$, for each $j \in [1, y_{i-1}]$. Let the replaced vertices form the set $W^{(i)} = \{v_r^{(i-1)}, v_{2r}^{(i-1)}, \ldots, v_{y_{i-1}r}^{(i-1)}\}$. The last $y_i - y_{i-1}$ edges in $T^{(i)}$ are formed by the vertices in $U^{(i)}$ and $W^{(i)}$: $t_j^{(i)} = u_{\lceil j/i \rceil}^{(i)} \cup \{w_j^{(i)}, w_{j+(y_i-y_{i-1})}^{(i)}, \ldots, w_{j+(r-2)(y_i-y_{i-1})}^{(i)}\}$, for each $j \in [y_{i-1}+1; y_i]$. The hypergraph $H^{(i)}$ is i-regular by induction: each vertex in $U^{(i)}$ is a root of a hyperstar with i edges, while every vertex in $H^{(i)} \backslash U^{(i)}$ has $i-1$ incident edges in $E^{(i-1)}$ and one incident edge in $T^{(i)}$. Then, the hypergraph $H(V,E) = H^{(\Delta)}$ is Δ-regular and r-uniform.

Now we show that *GreedyMAX* finds a cover S of size $\sum_{i=1}^{\Delta} x_i$ in H, while an optimal cover S^* in H is of size $|E|/\Delta$. Thus, the ratio between the sizes of the optimal independent set $I^* = V \backslash S^*$ and the greedy independent set $I = V \backslash S$ is the one defined in (12). Since the hypergraph $H = H^{(\Delta)}$ is Δ-regular, *GreedyMAX* might start by selecting all vertices in $U^{(\Delta)}$ and deleting all edges in $T^{(\Delta)}$. The remaining hypergraph is $H^{(\Delta-1)}$ and *GreedyMAX* might continue by selecting all vertices in $U^{(\Delta-1)}$ and deleting $T^{(\Delta-1)}$. Inductively, *GreedyMAX* might select all vertices in $U^{(\Delta)} \cup \ldots \cup U^{(1)}$ as a minimal cover S of size $\sum_{i=1}^{\Delta} ix_i$ and output the remaining $(r-1)x_1$ vertices as a maximal independent set I.

Let $z_1 = x_1$ and $z_i = x_i - z_{i-1}/i$ for any $i \in [2,\Delta]$. An optimal cover S^* includes all vertices from $U^{(1)}$ and the last z_i vertices from each $U^{(i)}$ for $i \in [2,\Delta]$ (note, that by definition x_i is multiple of any $j \in [i+1, \Delta]$, then z_i is also a multiple of any $j \in [i+1, \Delta]$). The vertices in $U^{(1)}$ cover all edges in $T^{(1)}$, and the first x_1 edges in every $T^{(i)}$ for $i \in [2, \Delta]$. By induction, the last z_i vertices in $U^{(i)}$ cover the remaining edges in $T^{(i)}$ and z_i edges in every $T^{(j)}$, where $j \in [i+1, \Delta]$. Consequently, all edges in H are covered by the vertices from S^*. Since H is Δ-regular and no two vertices from S^* appear in the same edge (by construction of H), S^* is an optimal cover of size $|E|/\Delta$. Then, an optimal independent set is of size $|I^*| = n - |E|/\Delta = n(r-1)/r$, because $|E| = n\Delta/r$ in Δ-regular r-uniform hypergraphs. Finally, the ratio in (12) can be simplified to $\frac{n}{rx_1} = \frac{n(r-1)}{r}\frac{1}{(r-1)x_1}$, which is exactly $|I^*|/|I|$. □

3.4 Partitioning

The idea of the partitioning approach is to split a given hypergraph into k induced subhypergraphs so that MIS can be solved optimally on each subhypergraph in polynomial time. This is based on the strategy of [13] for ordinary graphs. Note, that the largest of the solutions on the subhypergraphs is a k-approximation of MIS, since the size of any optimal solution is at most the sum of the sizes of the largest weak independent sets on each subhypergraph.

We extend a partitioning lemma of Lovász [21] to the hypergraph case.

Lemma 3.17 *The vertices of a given hypergraph can be partitioned into $\lceil(\Delta+1)/3\rceil$ sets, each inducing a subhypergraph of maximum degree at most two.*

Proof: Start with an arbitrary vertex partitioning into $\lceil(\Delta+1)/3\rceil$ sets. While a set contains a vertex v with degree more than two, move v to another set that properly contains at most two edges incident on v. Such a set exists, because otherwise the total number of edges incident on v would be at least $3\lceil(\Delta+1)/3\rceil \geq \Delta+1$. Any such move increases the number of edges between different sets, and so the process terminates with a partition where every vertex has at most two incident edges in its set. □

The method can be implemented in time $O(\sum_{e \in E} |e|)$ by using an initial greedy assignment as argued in [13].

Lemma 3.18 *Weighted MIS in hypergraphs of maximum degree two can be solved optimally in polynomial time.*

Proof: Given a hypergraph $H(V, E)$ we consider the dual hypergraph $H'(E, V)$, whose vertices $e_1, \ldots, e_m$ correspond to the edges of H and the edges $v_1, \ldots, v_n$ correspond to the vertices of H, i.e. $v_i = \{e_j : v_i \in e_j \text{ in } H\}$. The maximum edge size in H' equals to the maximum degree of H, thus H' is a graph, possibly with loops. A vertex cover in H is an edge cover in H' (where an edge cover in H' is defined as a subset of edges that touches every vertex in H'), and a minimum weighted edge cover in graphs can be found in polynomial time via maximum weighted matching [9]. All edges not in a minimum cover in H' correspond to the vertices in H that form a maximum weak independent set in H. $\square$

The following result is straightforward from Lemmas 3.17 and 3.18.

Theorem 3.19 *Weighted MIS can be approximated within* $\lceil(\Delta + 1)/3\rceil$ *in polynomial time.*

3.5 The GreedyMIN Algorithm

The *GreedyMIN* algorithm iteratively adds a vertex of minimum degree into the weak independent set and deletes it from the hypergraph. If the vertex deletion results in loops (edges containing only one vertex), then the algorithm also deletes the vertices with loops along with all edges incident on such vertices. The algorithm terminates when the vertex set is empty. In Figure 3.5 is the formal description of the algorithm.

```
ALGORITHM GreedyMIN (H)
    I = ∅
    While the vertex set is not empty
      Add a vertex v of minimum degree to I
      Delete v from H
      Delete all vertices with loops along with all edges incident on them from H
    Output I
```

Fig. 4. The algorithm *GreedyMIN*

Theorem 3.20 *The performance ratio of GreedyMIN is at most* $\Delta - 1$.

Proof: Let I and I^* be the greedy and the optimal solutions. We split the sequence of iterations of the algorithm into epochs, where a new epoch starts when the algorithm selects a vertex of degree Δ. Clearly, if the algorithm always selects a vertex of degree less than Δ, the whole sequence of iterations is just one epoch. Let I_t and I_t^* be the set of vertices from the greedy and the optimal solutions, respectively, deleted during epoch t. Then, $|I| = \sum_t |I_t|$ and $|I^*| = \sum_t |I_t^*|$. We show that $|I_t^*|/|I_t| \leq \Delta - 1$ for every epoch t.

Consider an iteration i in epoch t. The algorithm selects a vertex v_i, whose set of neighbors in 2-edges we denote by $N(v_i)$. The vertices of $N(v_i)$ are deleted in the iteration along with all incident edges. The maximum number of nodes removed in the iteration i that can belong to I_t^* is at most the degree of v_i. If i is the first iteration in t, then $d(v_i) = \Delta$; for any other iteration in the same epoch $d(v_i) < \Delta$ (by the definition of an epoch).

Suppose one of the deleted edges is incident on a vertex u outside of $N(v_i)$. Then, in iteration $i+1$, the vertex u will have degree at most $\Delta - 1$, and therefore, the degree of v_{i+1} is at most $\Delta - 1$. Thus, the iteration $i+1$ will be in the same epoch as i, and the maximum number of nodes removed in any such iteration that can belong to I_t^* is at most $\Delta - 1$.

The last iteration of an epoch occurs when a vertex v_j is chosen whose neighborhood is contained in $N(v_j) \cup \{v_j\}$. This neighborhood then forms a hyperclique, because any vertex in $N(v_j)$ has at least the degree of v_j and all its neighbors are contained in $N(v_j) \cup \{v_j\}$. Notice that we may assume without loss of generality that the hypergraph is *simple*, namely that no edge is a proper subset of any other edge. Therefore, since the degree of v_j is at most Δ, any edge of the hyperclique contains at most $\Delta - 1$ vertices, and the maximum number of nodes removed in this iteration that can belong to an optimal solution I_t^* is at most $\Delta - 2$.

We see that in any epoch t the maximum number of deleted vertices that belong to I_t^* is at most Δ in the first iteration, at most $\Delta - 2$ in the last iteration and at most $\Delta - 1$ in any intermediate iteration. Amortized, the maximum number of deleted vertices that belong to I_t^* in any iteration of epoch t is at most $\Delta - 1$, while exactly one deleted vertex belongs to I_t. Therefore, $|I_t^*|/|I_t| \leq \Delta - 1$ for every epoch t. □

Theorem 3.21 *The performance ratio of GreedyMIN is at least* $\Delta - 1$ *for* $\Delta = 3$ *and at least* $\Delta - 2 + \frac{2}{\Delta+1}$ *for any* $\Delta \geq 4$.

Proof: We consider two cases: $\Delta = 3$ and $\Delta \geq 4$, and describe hard hypergraphs for both cases. Let an *n-star* refer to a star with $n+1$ vertices.

Case I: $\Delta = 3$. For any $l \geq 2$ we construct a 3-regular hypergraph, composed of l 2-stars (see Fig. 5). For $i \in [1, l]$, each 2-star H_i has a root t_i and 2 endpoints v_i and u_i, connected to the root by the edges (t_i, v_i) and (t_i, u_i). The root t_i of each star H_i is connected to the endpoints of the preceding star by one edge (t_i, u_{i-1}, v_{i-1}) (the root of the last star is connected to the endpoints of the first star by an edge (t_l, u_1, v_1)). The endpoints of all stars are connected into one edge $(u_1, u_2, \ldots, u_l, v_1, v_2, \ldots, v_l)$.

Since the hypergraph is regular, the algorithm might start by selecting the root of the first star, adding it to the independent set and deleting it from the hypergraph. After this deletion, the endpoints of the second star have loops,

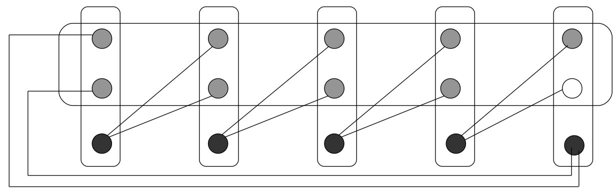

Fig. 5. Example of a hard 3-regular hypergraph for GreedyMIN, where the grey vertices represent an optimal solution, the black vertices represent the greedy solution.

and so the algorithm deletes the endpoints of the second star with all incident edges, reducing by one the degree of the endpoints of all other stars and the root of the second star. The algorithm proceeds this way, choosing all the roots of the stars for a solution of size l. On the other hand, an optimal solution is of size $l(\Delta-1)-1$ and includes the endpoints of all but one stars. Therefore, the performance ratio is $\rho = \Delta - 1 - \frac{1}{l}$, approaching $\Delta - 1$, when l is large.

Case II: $\Delta \geq 4$. We construct a Δ-regular hypergraph, composed of Δ blocks and a vertex s. For $i \in [1, l]$, each block is a Δ-star H_i with a root t_i and Δ endpoints $\{v_i^1, \ldots, v_i^\Delta\}$ connected to the root by Δ edges $\{(t_i, v_i^1), (t_i, v_i^2), \ldots, (t_i, v_i^\Delta)\}$. In each block the vertices $\{v_i^1, \ldots, v_i^{\Delta-1}\}$ are connected to the vertex s by a single edge $(s, v_i^1, \ldots, v_i^{\Delta-1})$; the vertex v_i^Δ is connected to the vertices $\{v_i^1, \ldots, v_i^{\Delta-1}\}$ by $\Delta - 1$ edges of cardinality $\Delta - 1$ each (see Fig. 6).

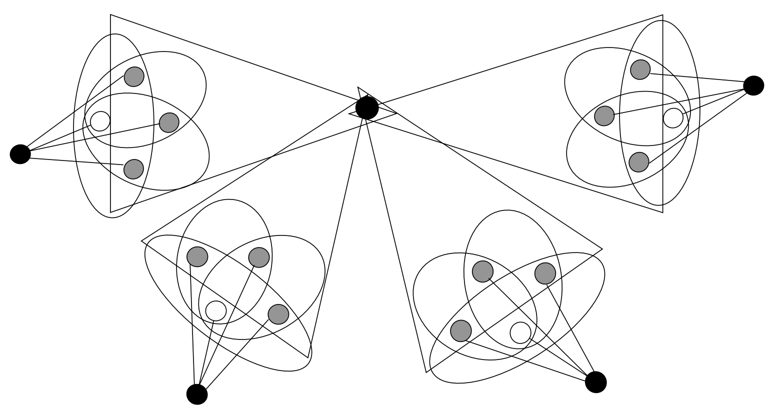

Fig. 6. Example of a hard 4-regular hypergraph for *GreedyMIN*, where the grey vertices represent an optimal solution, the black vertices represent the greedy solution.

The hypergraph is regular, and so the algorithm might start by selecting the vertex s. The deletion of s doesn't change the degree of the remaining vertices, because s has no incident 2-edges and the algorithm doesn't delete any edges. This leaves disjoint regular Δ-stars, where the greedy algorithm chooses only the roots of the stars for a solution of size $\Delta + 1$. On the other hand, an optimal solution is of size $\Delta(\Delta - 1)$ and includes $\Delta - 1$ endpoints from each star. Therefore, the performance ratio is $\rho = \frac{\Delta-1}{1+1/\Delta} = \Delta - 2 + \frac{2}{\Delta+1}$. □

Theorem 3.22 *In r-uniform Δ-regular hypergraphs GreedyMIN approaches the performance ratio of* $1 + \frac{\Delta-1}{r}$.

Proof: We assume that $r \geq 3$, because 2-uniform hypergraphs are ordinary graphs and the analysis of the greedy algorithm on graphs can be found in [14].

Given a hypergraph $H(V, E)$, let I be a weak independent set in H. We denote by I_{max} and I_{min} the largest and the smallest maximal weak independent sets in H. The performance ratio of any approximation algorithm for MIS is bounded by the maximum ratio between I_{max} and I_{min} taken over all hypergraphs:

$$\rho \leq \max_{\forall H} \frac{|I_{max}|}{|I_{min}|} . \tag{13}$$

Any minimal cover S in H is of size at least

$$|S| \geq \frac{|E|}{\Delta} = \frac{\Delta|V|}{r\Delta} = \frac{|V|}{r} , \tag{14}$$

where in the last equality we use the fact that the number of edges in r-uniform Δ-regular hypergraph is exactly $|E| = \frac{\Delta|V|}{r}$. It is also easy to prove that any minimal cover is of size at most:

$$|S| \leq \frac{\Delta|V|}{\Delta + r - 1} . \tag{15}$$

For the reader's convenience we cite here the proof of (15) from [2]. Since S is a minimal cover, for any vertex $v \in S$ there is at least one edge in E covered only by v. Consequently, each such edge includes $r - 1$ vertices from $V \backslash S$ and the total degree of vertices in $V \backslash S$ is at least $|S|(r - 1)$. On the other hand, the total degree of vertices in $V \backslash S$ is at most $\Delta(|V| - |S|)$. From $|S|(r - 1) \leq \Delta(|V| - |S|)$ the inequality (15) follows immediately.

Any vertex in V belongs either to a minimal cover or to a maximal weak independent set, then $|I| = |V| - |S|$. Consequently, any maximal weak independent set is of size at least:

$$|I_{min}| \geq |V| - \frac{\Delta|V|}{\Delta + r - 1} \tag{16}$$

and at most

$$|I_{max}| \leq |V| - \frac{|V|}{r} , \tag{17}$$

where the first inequality involves the upper bound on the size of a minimal cover in H from (15), and the second inequality uses the lower bound from (14).

Finally, combining together (13), (16) and (17) we obtain the upper bound on the performance ratio of any approximation algorithm for MIS:

$$\rho \leq \frac{1 - \frac{1}{r}}{1 - \frac{\Delta}{\Delta + r - 1}} = \frac{\Delta + r - 1}{r} = 1 + \frac{\Delta - 1}{r} \ . \tag{18}$$

For the lower bound, we construct a Δ-regular r-uniform hypergraph H composed of a hyperclique B on $\Delta + 1$ vertices and a set A of $r - 1$ vertices. The edges of the hyperclique are all possible Δ-combinations of $\Delta + 1$ vertices. Each vertex of the hyperclique except one is connected to the set A by one edge.

Since the hypergraph is regular, the *GreedyMIN* algorithm might start by selecting vertices in the set A. The deletion of the first $r-2$ vertices reduces the size of the incident edges from r to 2 and doesn't produce loops. The deletion of the last vertex in A creates loops on all vertices in B, and the algorithm deletes the set B and all incident edges. Thus, the greedy weak independent set includes $r - 1$ vertices from A and one vertex from B, while an optimal weak independent set includes Δ vertices from B and $r - 2$ vertices from A. The approximation ratio is then $\frac{r-2+\Delta}{r}$. □

4 Strong Independent Set

There are two greedy algorithms for the MSIS problem in hypergraphs. Both algorithms iteratively construct a maximal strong independent set by selecting vertices either of minimum degree (the GreedyD algorithm) or with fewest neighbors (the GreedyN algorithm).

Lemma 4.1 *Any maximal strong independent set is a Δ-approximation.*

Proof: Each node in the optimal solution is dominated by a node in the maximal solution, i.e. either by itself or by its neighbor. However, each node in the maximal solution can dominate at most Δ optimal vertices, as its neighborhood is covered by at most Δ edges, each containing at most one optimal vertex. □

Lemma 4.2 *There exist Δ-regular hypergraphs where the approximation ratio of GreedyD is Δ.*

Proof: For any $l \geq 2$ we construct the hypergraph $H_l(V, E)$, composed of a vertex s and l cliques on l vertices each. The vertex s is connected to the cliques by l edges, so that the i-th edge includes the vertex s and the i-th vertex from each clique.

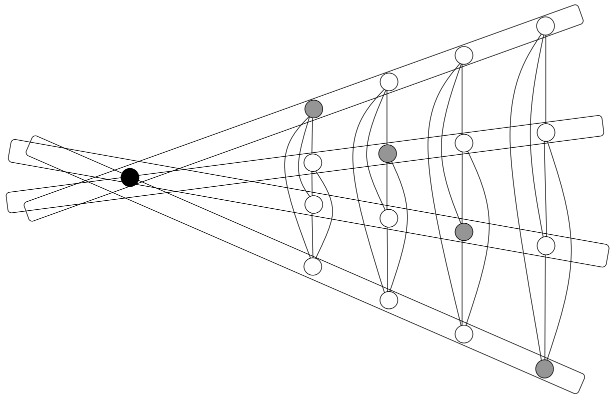

Fig. 7. Example of a hard 4-regular hypergraph for GreedyD, where the grey vertices represent an optimal solution, the black vertex represents the greedy solution.

Each vertex in the hypergraph has degree l, and so the hypergraph is regular with $\Delta = l$. The maximum strong independent set is of size l and includes the i-th vertex from the i-th clique. GreedyD is a non-deterministic algorithm: in the worst case the vertex s is selected first and no more vertices can be added to the solution. Thus, the performance ratio is Δ. $\square$

Lemma 4.3 *There exist Δ-regular hypergraphs where the performance ratio of GreedyN approaches Δ.*

Proof: For any $m \geq 2$ and $l \geq 2$ we construct the hypergraph $H_{m,l}(V, E)$, composed of m subgraphs on $3l$ vertices each. For $i \in [1, m]$, each subgraph H_i consists of a set U_i of l vertices and a complete bipartite graph (W_i, T_i) with $|W_i| = |T_i| = l$, without one matching. For each vertex in W_i there is an edge containing this vertex and the set U_i. All subgraphs are connected by one edge, containing all T sets.

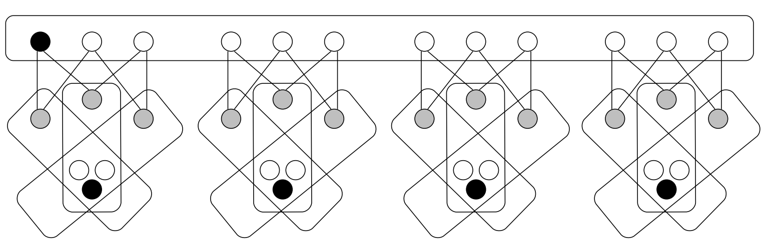

Fig. 8. Example of a hard 4-regular hypergraph for GreedyN, where the grey vertices represent an optimal solution, the black vertices represent the greedy solution.

Each vertex in the hypergraph has degree l, and so the hypergraph is regular with $\Delta = l$. We can easily verify that every vertex in U and W has the same number of neighbors, namely $2l - 1$, and every vertex in T has $l(m+1) - 2$ neighbors. In each subgraph H_i every vertex in U_i is a neighbor of $l-1$ vertices in U_i and l vertices in W_i; every vertex in W_i is a neighbor of l vertices in U_i and $l-1$ vertices in T_i; every vertex in T_i is a neighbor of $l-1$ vertices in W_i, $l-1$ vertices in T_i and $(m-1)l$ vertices in T-sets from the other $m-1$ subgraphs.

A maximum strong independent set is of size ml and includes all W sets. GreedyN is a non-deterministic algorithm, and so it might start by selecting a vertex from U_1, delete U_1 and W_1 from the subgraph and reduce the number of neighbors of any vertex in T_1 to $l(m-1)$. Since $m \geq 2$, the vertices in T_1 have at least the same number of neighbors as the vertices in any of the U and W sets of the remaining subgraphs. Thus, the algorithm might proceed by selecting a vertex from U_2 and so on until all U and W sets are deleted. From the remaining edge composed of all T sets, the algorithm adds only one vertex to the solution. Therefore, the greedy solution is of size $m+1$ and the performance ratio is approximately $l = \Delta$ provided m is large. □

Theorem 4.4 *In r-uniform hypergraphs the performance ratio of GreedyD and GreedyN is at most $\Delta - \frac{\Delta-1}{r}$.*

Proof: Let v_i be the vertex chosen by the algorithm (GreedyD or GreedyN) on the i-th iteration; d_i and n_i denote the degree and the number of neighbors of v_i, respectively. The greedy algorithm terminates when the vertex set is empty, say after t iterations:

$$\sum_{i=1}^{t}(n_i + 1) = n \ . \tag{19}$$

Since the vertex v_i has n_i neighbors, its degree is at least:

$$d_i \geq \frac{n_i}{r-1} \ . \tag{20}$$

Any neighbor v_j of v_i has at least the degree $\frac{n_i}{r-1}$. The reason is simple: in GreedyD the vertex v_i has the smallest degree, and so the degree of v_j is at least the degree of v_i; in GreedyN the vertex v_j has at least the same number of neighbors as v_i and consequently, it is degree is $d_j \geq \frac{n_j}{r-1} \geq \frac{n_i}{r-1}$. Then, the total sum of degrees of all vertices in the hypergraph equals to $\bar{d}n$:

$$\bar{d}n \geq \sum_{i=1}^{t}\left(d_i + \sum_{j=1}^{n_i} d_j\right) \geq \sum_{i=1}^{t}\left(\frac{n_i}{r-1} + \sum_{j=1}^{n_i}\frac{n_i}{r-1}\right)$$

$$= \frac{1}{r-1}\sum_{i=1}^{t} n_i(n_i+1) = \frac{1}{r-1}\left(\sum_{i=1}^{t}(n_i+1)^2 - n\right) \tag{21}$$

$$\geq \frac{1}{r-1}\left(\frac{n^2}{t} - n\right) \tag{22}$$

where in (21) we use Cauchy-Schwarz inequality [3]. From (22) we can derive the lower bound on the size of the greedy solution:

$$t \geq \frac{n}{\bar{d}(r-1)+1} .$$

Let δ be the minimum degree in a given hypergraph. Since the number of edges in r-uniform hypergraphs is $\bar{d}n/r$ and each edge includes at most one vertex from a maximum strong independent set, the size of any maximum strong independent set is at most:

$$\alpha \leq \frac{\bar{d}n}{\delta r} .$$

Then, the performance ratio of the greedy algorithm (GreedyD or GreedyN) is at most:

$$\rho = \max_{\forall \alpha, t}\frac{\alpha}{t} \leq \frac{\bar{d}}{\delta r}(\bar{d}(r-1)+1) .$$

Let k be such that $\bar{d} = k\Delta + (1-k)\delta$. Then, it is easy to verify that $f(k) = \frac{k\Delta+(1-k)\delta}{\delta r}((k\Delta+(1-k)\delta)(r-1)+1)$ is maximized when $\Delta = \delta$ or $k = 1$, i.e. in regular hypergraphs. $\square$

Theorem 4.5 *In r-uniform hypergraphs the performance ratio of GreedyD and GreedyN is at least $\Delta - \frac{\Delta-1}{r}$.*

Proof: We describe the construction for the GreedyD algorithm; for GreedyN it is similar. The hypergraph H is composed of r subgraphs on $\Delta r - \Delta + 1$ vertices each. The first $r-1$ subgraphs H_i are disjoint, each of them consists of a vertex s, a set A of Δ independent vertices and a set B of $\Delta(r-2)$ vertices. The r-th subgraph is connected to the first $r-1$ subgraphs and contains a vertex s and a set C of $\Delta(r-1)$ vertices. In each subgraph the vertex s is connected to all other vertices by Δ-edges: in the first $r-1$ subgraphs each such edge includes one vertex from A and $r-2$ vertices from B, while in the last subgraph each such edge includes $r-1$ C-vertices. In each of the first $r-1$ subgraphs there are also $\Delta-1$ edges incident on each vertex in A: half of these edges includes $(r-1)$ vertices from B, the other half of the edges includes

[3] Cauchy-Schwarz inequality for one dimensional space: $\sum_{i=1}^{n} x_i^2 \geq \frac{1}{n}\left(\sum_{i=1}^{n} x_i\right)^2$

$(r-3)$ vertices from B and two vertices from C. We can specify the edges such that all edges have the cardinality r and all vertices in the hypergraph have the same degree Δ.

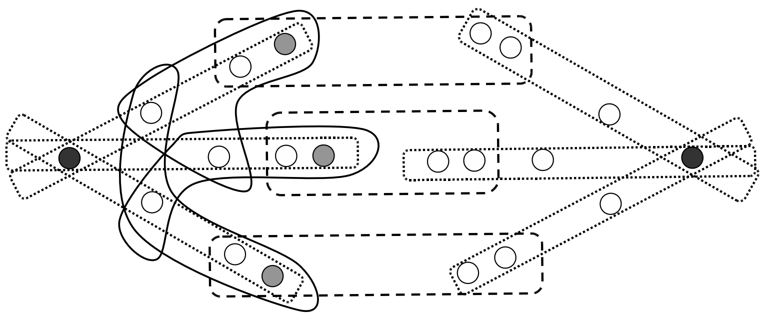

Fig. 9. Part of a hard 3-regular 4-uniform hypergraph for GreedyN, where one of the first $r-1$ subgraphs is connected to the last subgraph. The black vertices represent the s-vertices, the grey vertices represent the set A and the white vertices represent the sets B and C.

A maximum strong independent set is of size $(r-1)\Delta+1$ and consists of all A-sets and the vertex s from the last subgraph. The greedy algorithm might start by selecting the vertex s from the first subgraph and deleting the sets A and B in the first subgraph. This deletion reduces the size of one edge in the last subgraph by $r-2$ vertices, but doesn't reduce the degree of any of the remaining vertices. Thus, on the next iteration the greedy algorithm might repeatedly select vertices s from each subgraphs, and form a maximal strong independent set of size r. Therefore, the performance ratio is $\Delta - \frac{\Delta-1}{r}$. □

Remarks. We conjecture that it should be possible to prove that *GreedyMIN* have the worst performance ratio in Δ-regular r-uniform hypergraphs, and so the result of Theorem 3.22 applies to arbitrary r-uniform hypergraphs. In any case, the performance ratio of *GreedyMIN* in Δ-regular r-uniform hypergraphs is worse than the performance ratio *GreedyMAX* in r-uniform hypergraphs.

References

[1] C. Bazgan, J. Monnot, V. Paschos and F. Serrière, On the differential approximation of MIN SET COVER, *Theoretical Computer Science*, 332:497–513, 2005.

[2] C. Berge, Hypergraphs, North-Holland, 1989.

[3] P. Berman, A $d/2$ approximation for Maximum Weight Independent Set in d-claw free graphs, *Nordic Journal of Computing*, 7:178–184, 2000.

[4] P. Berman and T. Fujito, On Approximation Properties of the Independent Set Problem for Low Degree Graphs, *Theory Computing Systems*, 32(2):115–132, 1999.

[5] P. Berman and M. Fürer, Approximating maximum independent set in bounded degree graphs, In *Proc. 5th Ann. ACM-SIAM Symp. on Discrete Algorithms (SODA)*, 365–371, 1994.

[6] J. Cardinal, S. Fiorini and G. Joret, Tight Results on Minimum Entropy Set Cover, *Algorithmica*, 50(1):49–60, 2008.

[7] Y. Caro and Z. Tuza, Improved lower bounds on k-independence, *Journal of Graph Theory*, 15(1):99–107, 1991.

[8] V. Chvátal, A greedy heuristic for the set-covering problem, *Mathematics of Operations Research*, 4(3):233–235, 1979.

[9] J. Edmonds, Paths, trees and flowers, *Canadian Journal of Mathematics*, 17:449–467, 1965.

[10] U. Feige, A threshold of $\ln n$ for approximating set cover, *Journal of the ACM*, 45(4):634–652, 1998.

[11] U. Feige, L. Lovász and P. Tetali, Approximating min-sum set cover, *Algorithmica*, 40(4):219–234, 2004.

[12] M. M. Halldórsson, Approximations of independent sets in graphs, In *Proc. 1s st. Int. Workshop on Approximation Algorithms for Combinatorial Optimization (APPROX)*, LNCS 1441:1–13, 1998.

[13] M. M. Halldórsson and H.-C. Lau, Low-degree graph partitioning via local search with applications to Constraint Satisfaction, Max Cut, and 3-Coloring, *Journal of Graph Algorithms and Applications*, 1:1–13, 1997.

[14] M. M. Halldórsson and J. Radhakrishnan, Greed is good: Approximating independent sets in sparse and bounded-degree graphs, *Algorithmica*, 18(1):143–163, 1997.

[15] E. Halperin, Improved approximation algorithms for the vertex cover problem in graphs and hypergraphs, *SIAM Journal on Computing*, 31(5):1608–1623, 2002.

[16] E. Hazan, S. Safra and O. Schwartz, On the hardness of approximating k-dimensional matching, *Electronic Colloquium on Computational Complexity*, 10(020), 2003.

[17] Th. Hofmeister and H. Lefman, Approximating maximum independent sets in uniform hypergraphs, In *Proc. 23rd Int. Symposium on Mathematical Foundations of Computer Science (MFCS)*, LNCS 1450:562–570, 1998.

[18] C. A. J. Hurkens and 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(1):68–72, 1989.

[19] D. S. Johnson, Approximation algorithms for combinatorial problems, *Journal of Computer and System Sciences*, 9:256–278, 1974.

[20] M. Krivelevich, R. Nathaniel and B. Sudakov, Approximating coloring and maximum independent set in 3-uniform hypergraphs, *Journal of Algorithms*, 41(1):99–113, 2001.

[21] L. Lovász, On decomposition of graphs, *Stud. Sci. Math. Hung.*, 1:237–238, 1966.

[22] L. Lovász, On the ratio of optimal integral and fractional covers, *Discrete Mathematics*, 13:383–390, 1975.

[23] S. Sakai, M. Togasaki and K. Yamazaki, A note on greedy algorithms for the maximum weighted independent set problem, *Discrete Applied Mathematics*, 126(2-3):313–322, 2003.

[24] T. Thiele, A lower bound on the independence number of arbitrary hypergraphs, *Journal of Graph Theory*, 32:241–249, 1999.

[25] L. Trevisan, Non-approximability results for optimization problems on bounded degree instances, In *Proc. 33rd Ann. ACM symposium on Theory of computing (STOC)*, 453–461, 2001.

[26] S. Vishwanathan, Private communication, 1998.

[27] L. A. Wolsey, An analysis of the greedy algorithm for the submodular set covering problem, *Combinatorica*, 2(4):385–393, 1982.