Approximating Maximum Independent Sets in Uniform Hypergraphs*

Thomas Hofmeister / Hanno Lefmann
Lehrstuhl Informatik 2, Universität Dortmund, D-44221 Dortmund, Germany
{hofmeist,lefmann}@Ls2.cs.uni-dortmund.de

Abstract

We consider the problem of approximating the independence number and the chromatic number of k-uniform hypergraphs on n vertices. For fixed integers $k \geq 2$, we obtain for both problems that one can achieve in polynomial time approximation ratios of at most $O(n/(\log^{(k-1)} n)^2)$. This extends results of Boppana and Halldórsson [5] who showed for the graph case that an approximation ratio of $O(n/(\log n)^2)$ can be achieved in polynomial time. On the other hand, assuming $NP \neq ZPP$, one cannot obtain in polynomial time for the independence number and the chromatic number of k-uniform hypergraphs an approximation ratio of $n^{1-\epsilon}$ for fixed $\epsilon > 0$.

1 Introduction

For a given graph $G = (V, E)$ with vertex set V and edge set $E \subseteq [V]^2$, a subset $I \subseteq V$ is called *independent*, if I contains no edges. The *independence number* $\alpha(G)$ is the size of a largest independent set. Computing an independent set I with $|I| = \alpha(G)$ is an NP-hard problem. Therefore, polynomial time approximation algorithms for this problem have been investigated, cf. [1], [9]. For an optimization instance I, let $OPT(I)$ denote the value of the optimal solution and $A(I)$ the solution found by an algorithm A. The approximation ratio $AR(n)$ of algorithm A is defined by

$$\max_{I_n} \left\{ \frac{OPT(I_n)}{A(I_n)}, \frac{A(I_n)}{OPT(I_n)} \right\}$$

where the maximum is taken over all instances I_n of input size n (e.g., graphs on n vertices).

A straightforward greedy strategy for computing independent sets was studied by Halldórsson and Radhakrishnan [11] and approximation ratios in terms of the average degree $d = 2 \cdot |E|/|V|$ and the maximum degree Δ of the graph $G = (V, E)$ with $|V| = n$ were given, i.e., $AR(n) \leq (d + 2)/2$ and $AR(n) \leq (\Delta + 2)/3$, respectively.

*This research was supported by the Deutsche Forschungsgemeinschaft as part of the Collaborative Research Center "Computational Intelligence" (SFB 531).

For arbitrary graphs on n vertices an approximation ratio of $O(n/(\log n)^2)$ can be achieved in polynomial time as was shown by Boppana and Halldórsson [5]. They also showed that in graphs G on n vertices with $\alpha(G) \geq n/k + m$, where k is a fixed integer, one can find in polynomial time an independent set of size at least $\Omega(m^{1/(k-1)})$. This was improved by Alon and Kahale [2] who applied semidefinite programming in conjunction with the Lovász θ-number $\theta(G)$ of a graph G and they showed that if $\theta(G) \geq n/k + m$, then an independent set of size at least $\tilde{\Omega}(m^{3/(k+1)})$ can be found in randomized polynomial time.
As far as inapproximability results are concerned, the latest result is due to Håstad [12] who showed that for all $\epsilon > 0$, there can be no polynomial time algorithm with approximation ratio of $n^{1/2-\epsilon}$, unless P=NP. Under the assumption that $NP \neq ZPP$, the same holds even for an approximation ratio of $n^{1-\epsilon}$.
The corresponding problem for hypergraphs has been less studied. A *hypergraph* $\mathcal{H} = (V, \mathcal{E})$ is given by a set V of vertices and a set $\mathcal{E}$ of edges where $E \subseteq V$ for every edge $E \in \mathcal{E}$. A hypergraph is called *k-uniform* if $|E| = k$ for every edge $E \in \mathcal{E}$. A subset $I \subseteq V$ is called *independent* if I contains no edges from $\mathcal{H}$, i.e., $E \not\subseteq I$ for each edge $E \in \mathcal{E}$. The size of a largest independent set in $\mathcal{H}$ is called the *independence number* of $\mathcal{H}$ and is denoted by $\alpha(\mathcal{H})$.
Some results are concerned with finding in parallel a maximal independent set in a hypergraph, see e.g. [6], [15]. However, the sizes of maximal independent sets might be far off the size of a maximum independent set. In terms of the average degree $d^{k-1} := k \cdot |\mathcal{E}|/|V|$ of a k-uniform hypergraph $\mathcal{H} = (V, \mathcal{E})$, derandomizing a probabilistic argument of Spencer [19] yields a linear time algorithm with approximation ratio $O(d)$.
We consider here the problem of approximating maximum independent sets in arbitrary k-uniform hypergraphs. It turns out that one can achieve in polynomial time an approximation ratio of $O(n/(\log^{(k-1)} n)^2)$. Here, $\log^{(k)} n$ denotes the k-fold iterated logarithm $\log \cdots \log n$.
By a simple reduction from the case of graphs, inapproximability results of the order $\Omega(n^{1-\varepsilon})$ for each given $\varepsilon > 0$ can be derived.
In connection with the maximum independent set problem we also consider the problem of coloring the vertices of a k-uniform hypergraph $\mathcal{H}$ with as few colors as possible. Call a coloring of the vertices of $\mathcal{H}$ *proper* if no edge is monochromatic. Let $\chi(\mathcal{H})$ denote the *chromatic number* of a hypergraph $\mathcal{H}$, i.e., the minimum number of colors which are necessary to obtain a proper coloring of the vertices of $\mathcal{H}$. If $l \geq \chi(\mathcal{H})$, then the hypergraph $\mathcal{H}$ is called *l-colorable*.
For graphs, determining the chromatic number is an NP-hard problem. The currently best known polynomial time algorithm for coloring a 3-colorable graph on n vertices is by Blum and Karger [4] and uses $\tilde{O}(n^{3/14})$ colors. For graphs with larger chromatic number, the currently best known result is due to Karger, Motwani and Sudan [14] who showed that graphs on n vertices with maximum degree Δ can be colored in polynomial time using at most min $\{\tilde{O}(\Delta^{1-2/\chi(G)}), \tilde{O}(n^{1-3/(\chi(G)+1)})\}$

colors.
For arbitrary graphs on n vertices, the first polynomial time approximation algorithm for graph coloring was given by Johnson [13] with an approximation ratio of $O(n/\log n)$. This was improved by Wigderson in [20] to $O(n \cdot (\log\log n/\log n)^2)$ and further by Boppana and Halldórsson [5] to $O(n/(\log n)^2)$. The latest improvement is by Halldórsson [10] who gave a polynomial time algorithm to achieve an approximation ratio of $O(n \cdot (\log\log n)^2/(\log n)^3)$. Feige and Kilian [8] recently showed that assuming $NP \neq ZPP$, given any $\epsilon > 0$, there is no polynomial time algorithm which approximates the chromatic number within a factor $O(n^{1-\epsilon})$.
For the corresponding problem of coloring hypergraphs not much is known. In fact, the first polynomial time algorithm capable of coloring a 2-colorable k-uniform hypergraph on n vertices with a sublinear number of colors was given by Kelsen, Mahajan and Ramesh [16] recently. Their algorithm uses $O(n^{1-1/k} \cdot (\log n)^{1-1/k})$ colors.
Our algorithm works not only for 2-colorable k-uniform hypergraphs, but for k-uniform hypergraphs without any restriction on their chromatic number. It achieves an approximation ratio of $O(n/(\log^{(k-1)} n)^2)$ for the chromatic number. Thus, if applied to k–uniform hypergraphs which are l–colorable for a fixed $l > 0$, our algorithm also only uses a sublinear number of colors, though in the case $l = 2$, the bound from [16] is better.
Nevertheless, it seems that the technique from [16] can not be applied to hypergraphs with chromatic number larger than 2. In particular, it was shown in [16] by using semidefinite programming, that 2-colorable hypergraphs where each edge has cardinality at most 3 can be colored with at most $O(n^{2/9} \cdot (\log n)^{17/8})$ colors in polynomial time. In their arguments the assumption on 2-colorability was essential. The semidefinite programming approach does not seem to be applicable for k-uniform hypergraphs where $k \geq 4$, cf. [16].
Our approximation ratio $O(n/(\log^{(k-1)} n)^2)$ is sublinear, but still rather large. Again, one should note that (assuming $NP \neq ZPP$) there is no polynomial time algorithm which approximates the chromatic number of hypergraphs on n vertices within a factor $O(n^{1-\epsilon})$ for every given $\epsilon > 0$.

2 An Approximation Algorithm

In addition to the notions given in the introduction, we also use the notion of a *clique* in a k-uniform hypergraph which is a complete subhypergraph of $\mathcal{H}$, i.e., contains, say, l vertices and all $\binom{l}{k}$ edges. Note that an approximation algorithm for computing independent sets can always be applied to the complement of a hypergraph to obtain an approximation algorithm for cliques, with the same approximation ratio.
The case $k = 2$, i.e., graphs, has been treated by Boppana and Halldórsson [5], who described the following algorithm 2–Ramsey.

Algorithm 2–Ramsey: Transform the graph into a binary tree, its vertices corresponding to the vertices of the graph, in such a way that for each vertex the set of its left descendants contains exactly the non-neighbours of v. Given a path to a leaf in this tree, the root and the set of vertices reached by a right edge form a clique. Furthermore, the root and the set of vertices reached by a left edge form an independent set. Compute the path with the largest number of right edges and the path with the largest number of left edges and the corresponding sets C and I. A Ramsey-type argument shows:

Theorem 2.1 [5] *In graphs on n vertices, algorithm 2–Ramsey returns an independent set I and a clique C such that $|I| \cdot |C| \geq 1/4 \cdot (\log n)^2$.*

In order to describe the algorithm for approximating maximum independent sets in k-uniform hypergraphs, we use some transformation which constructs $(k-1)$-uniform hypergraphs from k-uniform hypergraphs:

Definition 2.2 *Let $2 \leq g < k$, and let $\mathcal{H}_g$ and $\mathcal{H}_k$ be g– and k–uniform hypergraphs, respectively, both with the same totally ordered vertex set V.*
Then, $\mathcal{H}_k$ is the ordered canonical extension of $\mathcal{H}_g$ whenever the following holds for all k–element subsets $E = \{v_1, \ldots, v_k\}$ with $v_1 < \ldots < v_k$ of the vertices:

E is an edge in $\mathcal{H}_k$ if and only if $\{v_1, \ldots, v_g\}$ is an edge in $\mathcal{H}_g$.

Moreover, $\mathcal{H}_k$ is the unordered canonical extension of $\mathcal{H}_g$ whenever the following holds for all k–element subsets E of the vertices:

E is an edge in $\mathcal{H}_k$ if and only if there is some edge e in $\mathcal{H}_g$ such that $e \subseteq E$.

It is rather easy to show that if the k-uniform hypergraph $\mathcal{H}_k$ is the (unordered) canonical extension of some g-uniform hypergraph $\mathcal{H}_g$ with $g < k$, then all cliques of size at least k in $\mathcal{H}_g$ are also cliques in $\mathcal{H}_k$ and all independent sets of size at least k in $\mathcal{H}_g$ are also independent sets in $\mathcal{H}_k$. Define an algorithm *k–Ramsey* as follows:

Algorithm k–Ramsey: If $k = 2$, apply the procedure by Boppana and Halldórsson. Otherwise, compute some induced subhypergraph of $\mathcal{H}_k$ which is the ordered canonical extension of a $(k-1)$-uniform hypergraph $\mathcal{H}_{k-1}$ and which has $\Omega((\log n)^{1/(k-1)})$ vertices. Apply *$(k-1)$–Ramsey* to $\mathcal{H}_{k-1}$.
Return the clique C and the independent set I that *$(k-1)$–Ramsey* returns.

The crucial point is that $\mathcal{H}_{k-1}$ should have a number of vertices which is not too small. The fact that such a hypergraph exists, follows from the next lemma which is due to Erdös and Rado [7], see also [17].

Lemma 2.3 [7] *Let $k \geq 2$ be fixed. Given a k-uniform hypergraph $\mathcal{H}_k$ on n vertices, one can find in polynomial time a subset S of the vertices with $|S| =$*

$\Omega((\log n)^{1/(k-1)})$*, and a* $(k-1)$*-uniform hypergraph* $\mathcal{H}_{k-1}$ *such that the induced subhypergraph of* $\mathcal{H}_k$ *on* S *is the ordered canonical extension of* $\mathcal{H}_{k-1}$*.*

Proof. Let $A, B \subseteq \{1, \ldots, n\}$ be nonempty sets such that $\max A < \min B$. We say that (A, B) is *good* if the following holds:

For each $(k-1)$–element subset S of A and for all $x, y \in B$ it holds that $S \cup \{x\}$ is an edge in $\mathcal{H}_k$ if and only if $S \cup \{y\}$ is an edge in $\mathcal{H}_k$.

Assume that we are given a good pair (A, B). Set $A' := A \cup \{\min B\}$ and let $j := |A|$. We claim that there is a subset $B' \subseteq B$ such that (A', B') is good and such that

$$|B'| \geq (|B| - 1)/2^{\binom{j}{k-2}} . \tag{1}$$

To see this, consider all $(k-1)$–element subsets S' of A'. If S' does not contain $\min B$, then the goodness condition holds for S' since (A, B) is good.
There are $\binom{j}{k-2}$ subsets S_i of A' which have cardinality $(k-1)$ and which contain $\min B$. Mark each $b \in B \setminus \{\min B\}$ with a bitstring of length $\binom{j}{k-2}$ with the meaning that the i–th bit is 1 if $S_i \cup \{b\}$ is an edge in $\mathcal{H}_k$ and 0 otherwise. By the pigeonhole principle, there must be a subset $B' \subseteq B \setminus \{\min B\}$ of cardinality at least $(|B| - 1)/2^{\binom{j}{k-2}}$ in which all are marked with the same bitstring. Hence, (A', B') is good. It is clear that B' can be found in polynomial time. We apply the following procedure:

$A = \{1, \ldots, k-2\}; B = \{k-1, \ldots, n\}; j = k-2;$
while $B \neq \emptyset$ **do**
  compute B' from B as sketched above ;
  $A = A \cup \{\min B\}$; $B = B'$; $j = j + 1$;
end;
return A;

The induced subhypergraph of $\mathcal{H}_k$ on the returned vertex set A is the ordered canonical extension of a $(k-1)$–uniform hypergraph on the same vertex set.
It remains to estimate $|A|$. Before each execution of the while-loop, the cardinality of B is $\Omega(n/2^{\binom{j}{k-1}})$. This is clear at the first execution and after that, we have by (1):

$$|B'| = \Omega(n/(2^{\binom{j}{k-1}} \cdot 2^{\binom{j}{k-2}})) = \Omega(n/2^{\binom{j+1}{k-1}}) .$$

Thus, the while-loop is executed $\Omega((\log n)^{1/(k-1)})$ times which is also a lower bound for the cardinality of A. During the whole procedure, each edge is only considered a constant number of times which means that it can be run in polynomial time. $\square$

Theorem 2.4 *Let $k \geq 2$ be a fixed integer. Given a k-uniform hypergraph $\mathcal{H}_k$ on n vertices, algorithm k–Ramsey returns in polynomial time a clique C and an independent set I such that $|C| \cdot |I| \geq c \cdot (\log^{(k-1)} n)^2$ for some constant $c > 0$.*

Proof. By induction. For $k = 2$, see Theorem 2.1. In the induction step, algorithm $(k-1)$–*Ramsey* is applied to a hypergraph with $\Omega((\log n)^{1/(k-1)})$ vertices. We then apply the induction hypothesis. □

Algorithm k–Ramsey guarantees that we either find a reasonable large clique or an independent set in a given k-uniform hypergraph. If however the hypergraph does not contain any large clique, then the size of an independent set should be large. The idea, as in [5], is to remove independent sets.

In step i, algorithm k–Ramsey is applied and returns a clique C_i and an independent set I_i. Remove the vertices from I_i and all incident edges from the hypergraph. This is repeated until no vertex is remaining. Let C be the clique C_i of maximum cardinality and let n_i denote the number of remaining vertices before step i.
Note that the removed independent sets I_i, $i = 1, \ldots, COL$, yield a proper coloring of the original hypergraph.
The following lemma was used in [5] for graphs, it also holds for hypergraphs and an implicit proof of it can be found in [18], Section 1.4.

Lemma 2.5 *If one can find an independent set of size at least $f(i)$ in k-uniform hypergraphs on i vertices, where $f(i)$ is non-decreasing and $f(i) > 0$, then one can find a proper coloring for a hypergraph on n vertices with at most $\sum_{i=1}^{n} 1/f(i)$ colors.*

The sets C_i and I_i satisfy $|C_i| \cdot |I_i| \geq c \cdot (\log^{(k-1)} n_i)^2$, i.e., $f(i) \geq c \cdot (\log^{(k-1)} i)^2/|C|$. By Lemma 2.5, we have (for $n \geq n_0$):

$$COL \leq \sum_{i=1}^{n} \frac{1}{c} \cdot \frac{|C|}{(\log^{(k-1)} i)^2} \leq \frac{2}{c} \cdot \frac{n \cdot |C|}{(\log^{(k-1)} n)^2} .$$

Let $\mathrm{cl}(\mathcal{H})$ denote the size of a largest clique in $\mathcal{H}$. Noting that $\mathrm{cl}(\mathcal{H}) \leq (k-1) \cdot \chi(\mathcal{H})$, one obtains

$$\frac{cl(\mathcal{H})}{k-1} \leq \chi(\mathcal{H}) \leq COL \leq \frac{2}{c} \cdot \frac{n \cdot |C|}{(\log^{(k-1)} n)^2} \leq \frac{2}{c} \cdot \frac{n \cdot cl(\mathcal{H})}{(\log^{(k-1)} n)^2} \leq c_k \cdot \frac{n \cdot \chi(\mathcal{H})}{(\log^{(k-1)} n)^2} . \quad (2)$$

From this chain of inequalities, one obtains the upper bounds for $cl(\mathcal{H})/|C|$ as well as for $COL/\chi(\mathcal{H})$ expressed in the following theorem:

Theorem 2.6 *Let $k \geq 2$ be a fixed integer. There is a polynomial time algorithm which approximates the independence number $\alpha(\mathcal{H})$, the clique number $\mathrm{cl}(\mathcal{H})$ and the chromatic number $\chi(\mathcal{H})$ of a k-uniform hypergraph $\mathcal{H}$ on n vertices with approximation ratio $O(n/(\log^{(k-1)} n)^2)$.*

One might consider the question how well the algorithm is performing on random hypergraphs. The answer is that unfortunately, the approximation ratio which we can guarantee is only slightly better than the trivial algorithms which pick one vertex as the independent set and color the graph with n colors.
Consider for example random k-uniform hypergraphs on n vertices where edges are present with probability 1/2 independently of each other. For such a random hypergraph, one can show that almost always $\mathrm{cl}(\mathcal{H}), \alpha(\mathcal{H}) = (1+o(1)) \cdot (k! \cdot \log n)^{\frac{1}{k-1}}$. Moreover, almost always $\chi(\mathcal{H})$ satisfies $\chi(\mathcal{H}) = (1+o(1)) \cdot n/(k! \cdot \log n)^{\frac{1}{k-1}}$. This can be seen by using similar techniques as in the case of graphs, cf. [3].
Considering the product

$$\frac{cl(\mathcal{H})}{|C|} \cdot \frac{COL}{\chi(\mathcal{H})} \leq \frac{cl(\mathcal{H})}{|C|} \cdot c_k \cdot \frac{n \cdot |C|}{(\log^{(k-1)} n)^2} \cdot \frac{1}{\chi(\mathcal{H})},$$

we see that it is almost always bounded by

$$O\left(\frac{(\log n)^{\frac{2}{k-1}}}{(\log^{(k-1)} n)^2}\right).$$

Note that for $k = 2$, this is a constant, while for $k \geq 3$, it is growing with n.
The trivial strategies on the other hand almost always lead to a product of the approximation ratios of $O((\log n)^{2/(k-1)})$.

3 Excluding l-Cliques

Very often, the approximation ratio can be improved if we know that our input graphs are taken from a certain subset of all graphs only. A particularly interesting subset which has often been considered in the literature is the set of all graphs which for some fixed $l \geq 2$ do not contain any clique on l vertices ("l–clique").
Boppana and Halldórsson observed in [5] that if a graph G on n vertices does not contain any l-clique for some $l \leq 2 \log n$, then G contains an independent set of size at least $\Omega(l \cdot n^{1/(l-1)})$, and for fixed $l \geq 2$ such an independent set can be found in polynomial time. They also showed that this implies that in graphs G on n vertices with $\alpha(G) \geq n/l + m$, where $l \geq 3$ is fixed, one can always find in polynomial time an independent set of size at least $\Omega(m^{1/(l-1)})$.
If we are given a hypergraph without l–cliques, then the reduction technique used in the algorithm k–Ramsey also guarantees that the graph to which we apply 2–Ramsey is also without l–cliques, hence instead of using 2–Ramsey in the final step of k–Ramsey, any of the procedures mentioned above can also be used.
As an example, one obtains

Theorem 3.1 *In every k–uniform hypergraph on n vertices which does not contain any l–cliques (for some fixed l), one can find in polynomial time an independent set of size at least* $\Omega((\log^{(k-2)} n)^{1/(2(l-1))})$.

4 Negative Results

The negative results follow from a straightforward transformation from the negative results for graphs. We list those results here for completeness. They follow from considering the unordered canonical extensions.

Theorem 4.1 *Let $k \geq 2$ be a fixed integer and let $\epsilon > 0$ be fixed. One cannot approximate in polynomial time the size of a maximum independent set (or clique) in k-uniform hypergraphs on n vertices within a factor of $O(n^{1-\epsilon})$, unless $NP = ZPP$.*

Proof. For k=2, the assertion holds by the result of Håstad [12]. Now assume $k > 2$ and that we had an approximation algorithm with ratio $O(n^{1-\epsilon})$ for k-uniform hypergraphs. Construct an approximation algorithm for graphs with the same ratio as follows: First, we check in polynomial time $O(n^k)$ whether the independence number of a graph G is smaller than k. If this is the case, then we can compute the independence number exactly. Otherwise, let $\mathcal{H}_k$ be the k-uniform hypergraph which is the unordered canonical extension of the graph G. Then it is rather easy to show that G and $\mathcal{H}_k$ have the same independence number. □

We apply a similar transformation to the approximation of the chromatic number:

Theorem 4.2 *Let $k \geq 2$ be a fixed integer and let $\epsilon > 0$ be fixed. One cannot approximate in polynomial time the chromatic number of a k-uniform hypergraph on n vertices within a factor of $O(n^{1-\epsilon})$, unless $NP = ZPP$.*

Proof. For k=2, the assertion holds by the result of Feige and Kilian [8]. For $k \geq 3$, consider the unordered canonical extension $\mathcal{H}_k$ of a given graph G. Then,

$$\chi(\mathcal{H}_k) \leq \chi(G) \leq (k-1) \cdot \chi(\mathcal{H}_k) \ .$$

The first inequality holds since every proper coloring of G is a proper coloring of $\mathcal{H}_k$ and for the second inequality observe that if we have a coloring of $\mathcal{H}_k$ with color classes $C_1, \ldots, C_l$, then for all $|C_i| \geq k$, we can color the corresponding vertices of G with the same color. If, however $|C_i| < k$, then we color the corresponding vertices in G by $|C_i|$ distinct colors. An approximation algorithm for k-uniform hypergraphs can hence be turned into an algorithm for graphs with an approximation ratio which is only worse by at most the constant factor $(k-1)$. □

5 Final Remarks and Questions

Can the approximation algorithm from [10] replace algorithm 2–Ramsey as the underlying procedure in an approximation algorithm for k–uniform hypergraphs?

Is it possible to improve the approximation ratio for the coloring problem or the maximum independent set problem in k-uniform hypergraphs to $o(n/(\log^{(k-1)} n)^2)$? It might also be interesting to investigate the approximation ratio with respect to polynomial time algorithms for the maximum independent set problem in l–clique free graphs or hypergraphs on n vertices. The algorithm from [5] can be seen as an approximation algorithm for graphs with approximation ratio of $O(n^{1-\frac{1}{l-1}})$ which shows that the inapproximability results from [12] do not carry over to this case, when l is fixed.

References

[1] N. Alon, L. Babai and A. Itai, *A Fast and Simple Randomized Parallel Algorithm for the Maximal Independent Set Problem,* Journal of Algorithms 7, 1986, 567-583.

[2] N. Alon and N. Kahale, *Approximating the Independence Number via the θ-Function*, Mathematical Programming, to appear.

[3] N. Alon and J. Spencer, *The Probabilistic Method,* Wiley & Sons, New York, 1992.

[4] A. Blum and D. Karger, *An $\tilde{O}(n^{3/14})$-coloring Algorithm for 3-colorable Graphs,* Information Processing Letters 61, 1997, 49-53.

[5] R. Boppana and M. Halldórsson, *Approximating Maximum Independent Sets by Excluding Subgraphs,* BIT 32, 1992, 180-196, also in: Proc. 2nd Scandinavian Workshop on Algorithm Theory (SWAT), Springer, LNCS 447, 1990, 13-25.

[6] E. Dahlhaus, M. Karpinski and P. Kelsen, *An Efficient Parallel Algorithm for Computing a Maximal Independent Set in a Hypergraph of Dimension 3,* Information Processing Letters 42, 1992, 309-313.

[7] P. Erdös and R. Rado, *Combinatorial Theorems on Classification of Subsets of a Given Set,* Proceedings London Mathematical Society 2, 1952, 417-439.

[8] U. Feige and J. Kilian, *Zero Knowledge and the Chromatic Number,* Proc. 11th IEEE Conference on Computational Complexity, 1996, 278-287.

[9] M. Goldberg and T. Spencer, *An Efficient Parallel Algorithm that Finds Independent Sets of Guaranteed Size,* SIAM Journal of Discrete Mathematics 6, 1993, 443-459.

[10] M. Halldórsson, *A Still Better Performance Guarantee for Approximate Graph Coloring,* Information Processing Letters 45, 1993, 19-23.

[11] M. Halldórsson and J. Radhakrishnan, *Greed is Good: Approximating Independent Sets in Sparse and Bounded-degree Graphs*, Proc. 26th ACM Symposium on the Theory of Computing (STOC), 1994, 439-448.

[12] J. Håstad, *Clique is Hard to Approximate Within* $n^{1-\epsilon}$, Proc. 37th IEEE Symposium on Foundations of Computer Science (FOCS), 1996, 627-636.

[13] D. S. Johnson, *Worst Case Behaviour of Graph Coloring Algorithms,* Proc. 5th Southeastern Conference on Combinatorics, Graph Theory and Computing, Congressus Numerantium X, 1974, 513-527.

[14] D. Karger, R. Motwani and M. Sudan, *Approximate Graph Coloring by Semidefinite Programming,* Proc. 35th IEEE Symposium on Foundations of Computer Science (FOCS), 1994, 2-13.

[15] P. Kelsen, *On the Parallel Complexity of Computing a Maximal Independent Set in a Hypergraph,* Proc. 24th ACM Symposium on the Theory of Computing (STOC), 1992, 339-350.

[16] P. Kelsen, S. Mahajan and H. Ramesh, *Approximate Hypergraph Coloring,* Proc. 6th Scandinavian Workshop on Algorithm Theory (SWAT), Springer, 1996, 41-52.

[17] J. Nešetřil, *Ramsey Theory,* in: Handbook of Combinatorics Vol II, eds. R. L. Graham, M. Grötschel and L. Lovász, North-Holland, 1995, 1331-1403.

[18] R. Motwani, P. Raghavan, *Randomized Algorithms,* Cambridge University Press, 1995.

[19] J. Spencer, *Turán's Theorem for k-Graphs,* Discrete Mathematics 2, 1972, 183–186.

[20] A. Wigderson, *Improving the Performance Guarantee for Approximate Graph Coloring,* Journal of the ACM 30, 1983, 729-735.