Approximate coloring of uniform hypergraphs (Extended abstract)

Michael Krivelevich[1] and Benny Sudakov[2]

[1] School of Mathematics, Institute for Advanced Study, Princeton, NJ 08540. Email: mkrivel@math.ias.edu. Research supported by an IAS/DIMACS Postdoctoral Fellowship.

[2] Department of Mathematics, Raymond and Beverly Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv, Israel. Email: sudakov@math.tau.ac.il.

Abstract. We consider an algorithmic problem of coloring r-uniform hypergraphs. The problem of finding the exact value of the chromatic number of a hypergraph is known to be NP-hard, so we discuss approximate solutions to it. Using a simple construction and known results on hardness of graph coloring, we show that for any $r \geq 3$ it is impossible to approximate in polynomial time the chromatic number of r-uniform hypergraphs on n vertices within a factor $n^{1-\epsilon}$ for any $\epsilon > 0$, unless $NP \subseteq ZPP$. On the positive side, we present an approximation algorithm for coloring r-uniform hypergraphs on n vertices, whose performance ratio is $O(n(\log\log n)^2/(\log n)^2)$. We also describe an algorithm for coloring 3-uniform 2-colorable hypergraphs on n vertices in $\tilde{O}(n^{9/41})$ colors, thus improving previous results of Chen and Frieze and of Kelsen, Mahajan and Ramesh.

1 Introduction

A *hypergraph* H is an ordered pair $H = (V, E)$, where V is a finite nonempty set (the set of *vertices*) and E is a collection of distinct nonempty subsets of V (the set of *edges*). H has *dimension* r if $\forall e \in E(H)$, $|e| \leq r$. If all edges have size exactly r, H is called *r-uniform.* Thus, a 2-uniform hypergraph is just a graph. A set $I \subseteq V(H)$ is called *independent* if U spans no edges of H. The maximal size of an independent set in H is called the *independence number* of H and is denoted by $\alpha(H)$. A *k-coloring* of H is a mapping $f : V(H) \to \{1, \ldots, k\}$ such that no edge of H (besides singletons) has all vertices of the same color. Equivalently, a k-coloring of H is a partition of the vertex set $V(H)$ into k independent sets. The *chromatic number* of H, denoted by $\chi(H)$ is the minimal k, for which H admits a k-coloring.

In this paper we consider an algorithmic problem of coloring r-uniform hypergraphs, for given and fixed value of $r \geq 2$. The special case $r = 2$ (i.e. the case of graphs) is relatively well studied and many results have been obtained in both positive (that is, good approximation algorithms, see e.g. [13], [19], [3], [11], [4], [14], [5]) and negative (that is, by showing the hardness of approximating the chromatic number under some natural complexity assumptions, see e.g. [16], [10]) directions. We will briefly survey these developments in the subsequent sections of the paper. However, much less is known about the general case. Lovász [17] showed that it is NP-hard to determine whether a 3-uniform hypergraph is 2-colorable. Additional results on complexity of hypergraph coloring were obtained in [18], [8], [7]. These hardness results give rise to attempts of developing algorithms for *approximate* uniform hypergraph coloring. The first non-trivial case of approximately coloring 2-colorable hypergraphs has recently been considered in papers of Chen and Frieze [9] and of Kelsen, Mahajan and Ramesh ([15], a journal version appeared in [1]). Both papers arrived independently to practically identical results. They presented an algorithm for coloring a 2-colorable r-uniform hypergraph in $O(n^{1-1/r})$ colors, using an idea closely related to the basic idea of Wigderson's coloring algorithm [19]. Another result of the above mentioned two papers is an algorithm for coloring 3-uniform 2-colorable hypergraphs in $\tilde{O}(n^{2/9})$ colors. The latter algorithm exploits the semidefinite programming approach, much in the spirit of the Karger-Motwani-Sudan coloring algorithm [14]. Nothing seems to have been known about general approximate coloring algorithms (that is, when the chromatic number of a hypergraph is not give in advance) and also about the hardness of this approximation problem.

This paper is aimed to (try to) fill a gap between the special case of graphs ($r = 2$) and the case of a general r. We present results in both negative and positive directions. In Section 2 we prove, using corresponding graph results, that unless $NP \subseteq ZPP$, for any fixed $r \geq 3$, it is impossible to approximate the chromatic number of r-uniform hypergraphs on n vertices in polynomial time within a factor of $n^{1-\epsilon}$, for any $\epsilon > 0$.

In Section 3 we present an approximation algorithm for coloring r-uniform hypergraphs on n vertices, whose performance guarantee is $O(n(\log\log n)^2/(\log n)^2)$, thus matching the approximation ratio of Wigderson's algorithm [19]. This algorithm is quite similar in a spirit to the algorithm of Wigderson, though technically somewhat more complicated.

In Section 4 we consider the special case of 3-uniform 2-colorable hypergraphs. We develop an algorithm whose performance guarantee for hypergraphs on n vertices is $\tilde{O}(n^{9/41})$, thus slightly improving over the result of Chen and Frieze [9] and of Kelsen, Mahajan and Ramesh [15]. Similarly to the improvement of Blum and Karger [5] of the Karger-Motwani-Sudan algorithm for coloring 3-colorable graphs, an improvement here is achieved by gaining some advantage in the case of dense hypergraphs. Final Section 5 is devoted to concluding remarks.

All logarithms are natural unless written explicitly otherwise.

2 Hardness of approximation

Several results on hardness of calculating exactly the chromatic number of r-uniform hypergraphs have been known previously. Lovász [17] showed that it is NP-complete to decide whether a given 3-uniform hypergraph H is 2-colorable. Phelps and Rödl proved in [18] that it is NP-complete to check k-colorability of r-uniform hypergraphs for all $k, r \geq 3$, even when restricted to linear hypergraphs. Brown and Corneil [8] presented a polynomial transformation from k-chromatic graphs to k-chromatic r-uniform hypergraphs. Finally, Brown showed in [7] that, unless $P = NP$, it is impossible to check in polynomial time 2-colorability of r-uniform hypergraphs for any $r \geq 3$.

However, we are not aware about any result showing that it is also hard to *approximate* the chromatic number of r-uniform hypergraphs, where $r \geq 3$. For the graph case ($r = 2$), Feige and Kilian showed in [10], using the result of Håstad [12], that if NP does not have efficient randomized algorithms, then there is no polynomial time algorithm for approximating the chromatic number of an n vertex graph within a factor of $n^{1-\epsilon}$, for any fixed $\epsilon > 0$.

In this section we present a construction for reducing the approximate graph coloring problem to approximate coloring of r-uniform hypergraphs, for any $r \geq 3$. Using this construction and the above mentioned result by Feige and Kilian we will be able to deduce hardness results in the hypergraph case.

Let $r \geq 3$ be a fixed uniformity number. Suppose we are given a graph $G = (V, E)$ on $|V| = n \geq r$ vertices with chromatic number $\chi(G) = k$. Define an r-uniform hypergraph $H = (V, F)$ in the following way. The vertex set of H is identical to that of G. For every edge $e \in E$ and for every $(r-2)$-subset $V_0 \subseteq V \setminus e$ we include the edge $e \cup V_0$ in the edge set F of H. If $F(H)$ contains multiple edges, we leave only one copy of each edge. The obtained hypergraph H is r-uniform on n vertices. Now we claim that $k/(r-1) \leq \chi(H) \leq k$. Indeed, a k-coloring of G is also a k-coloring of H, implying the upper bound on $\chi(H)$. To prove the lower bound, let $f : V \to \{1, \ldots, k'\}$ be a k'-coloring of H. Let G_0 be a subgraph of G, whose vertex set is V and whose edge set is composed of all these edges of G that are monochromatic under f. It is easy to see that the degree of every vertex $v \in V$ in G_0 is at most $r-2$ (otherwise the union of the edges of G_0 incident with v would form a monochromatic edge in H). Thus G_0 is $(r-1)$-colorable. We infer that the edge set $E(G)$ of G can be partitioned into two subsets $E(G) \setminus E(G_0)$ and $E(G_0)$ such that the first subset forms a k'-colorable graph, while the second one is $(r-1)$-colorable. Then G is $k'(r-1)$-colorable, as we can label each vertex by a pair whose first coordinate is its color in a k'-coloration of the first subgraph, and the second coordinate comes from an $(r-1)$-coloration of the second subgraph. Therefore G and H as defined above have the same number of vertices, and their chromatic numbers have the same order. Applying now the result of Feige and Kilian [10], we get the following theorem.

Theorem 1. *Let $r \geq 3$ be fixed. If $NP \not\subseteq ZPP$, it is impossible to approximate the chromatic number of r-uniform hypergraphs on n vertices within a factor of $n^{1-\epsilon}$ for any fixed $\epsilon > 0$ in time polynomial in n.*

3 A general approximation algorithm

In this section we present an approximation algorithm for the problem of coloring r-uniform hypergraphs, for a general fixed $r \geq 3$.

Let us start with describing briefly the history and state of the art of the corresponding problem of approximate graph coloring ($r = 2$). The first result on approximate graph coloring belongs to Johnson [13], who in 1974 proposed an algorithm with approximation ratio of order $n/\log n$, where $n = |V(G)|$. The next step was taken by Wigderson [19], whose algorithm achieves approximation ratio $O(n(\log\log n)^2/(\log n)^2)$. The main idea of Wigderson's algorithm was quite simple: if a graph is k-colorable then the neighborhood $N(v)$ of any vertex $v \in V(G)$ is $(k-1)$-colorable, thus opening a way for recursion. Berger and Rompel [3] further improved Wigderson's result by a factor of $\log n/\log\log n$. They utilized the fact that if G is k-colorable then one can find efficiently a subset S of a largest color class which has size $|S| \geq \log_k n$ and neighborhood $N(S)$ of size at most $n(1-1/k)$. Repeatedly finding such S and deleting it and its neighborhood leads to finding an independent set of size $(\log_k n)^2$. Finally, Halldórsson [11] came up with an approximation algorithm that uses at most $\chi(G)n(\log\log n)^2/(\log n)^3$ colors, currently best known result. His contribution is based on Ramsey-type arguments for finding a large independent set from his paper with Boppana [6]. Both papers [3] and [11] proceed by repeatedly finding a large independent set, coloring it by a fresh color and discharging it - quite a common approach in graph coloring algorithms. We will also adopt this strategy. It is worth noting here that one cannot hope for a major breakthrough in this question due to the hardness results mentioned in Section 2.

Unfortunately, most of the ideas of the above discussed papers do not seem to be applicable to the hypergraph case (i.e., when $r \geq 3$). It is not clear how to define a notion of the neighborhood of a subset in order to apply the Berger-Rompel approach. Also, bounds on the hypergraph Ramsey numbers are too weak to lead to algorithmic applications in the spirit of [6], [11]. However, something from the graph case can still be rescued. Both papers [9] and [15], dealing with the case of 2-colorable hypergraphs, noticed that the main idea behind Wigderson's algorithm is still usable for the hypergraph case. Let us describe now the main instrument of these papers, playing a key role in our arguments as well. For a hypergraph $H = (V, E)$ and a subset of vertices $S \subseteq V$, let $N(S) = \{v \in V : S \cup \{v\} \in E\}$. The following procedure is used in both papers [9] and [15].

Procedure *Reduce*(H, S)
Input: A hypergraph $H = (V, E)$ and a vertex subset $S \subseteq V$.
Output: A hypergraph $H' = (V, E')$.
1. Delete from E the set of edges $\{S \cup \{v\} : v \in N(S)\}$;
2. Add to E an edge S, denote the resulting hypergraph by H'.

It is easy to see that this procedure has the following properties.

Proposition 1. *Let $H' = Reduce(H, S)$.*

1. *if U is an independent set in H', then U is independent in H;*
2. *If H is k-colorable and the induced subhypergraph $H[N(S)]$ is not $(k-1)$-colorable, then H' is k-colorable.*

Note that the above proposition replaces edges of H by an edge of smaller size. Therefore, in order to apply it we need to widen our initial task and instead of developing an algorithm for coloring r-uniform hypergraphs to present an algorithm for hypergraphs of dimension r. Based on Prop. 1, we can use a recursion on k for coloring k-colorable hypergraphs of dimension r. Indeed, if for some S the subset $N(S)$ is relatively large and $H[N(S)]$ is $(k-1)$-colorable, then applying recursion we can find a relatively large independent subset of $N(S)$. If $H[N(S)]$ is not $(k-1)$-colorable, we can use procedure *Reduce*(H, S) in order to reduce the total number of edges. Finally, when the hypergraph is relatively sparse, a large independent set can be found based on the following proposition.

Proposition 2. *Let $H = (V, E)$ be a hypergraph of dimension $r \geq 2$ on n vertices without singletons. If every subset $S \subset V$ of size $1 \leq |S| \leq r-1$ has a neighborhood $N(S)$ of size $|N(S)| \leq t$, then H contains an independent set U of size $|U| \geq \frac{1}{4}(n/t)^{1/(r-1)}$, which can be found in time polynomial in n.*

Proof. For every $2 \leq i \leq r$, Let E_i be the set of all edges of size i in H. Then $E = \bigcup_{i=2}^{r} E_i$. By the assumptions of the proposition we have

$$|E_i| \leq \frac{\binom{n}{i-1}t}{\binom{i}{i-1}} \leq n^{i-1}t \ .$$

Choose a random subset V_0 of V by taking each $v \in V$ into V_0 independently and with probability $p_0 \geq 1/n$, where the exact value of p_0 will be chosen later. Define random variables X, Y by letting X be the number of vertices in V_0 and letting Y be the number of edges spanned by V_0. Then

$$E[X] = np_0, \qquad E[Y] = \sum_{i=2}^{r} |E_i| p_0^i \leq \sum_{i=2}^{r} n^{i-1} t p_0^i \leq (r-1) n^{r-1} p_0^r t \ .$$

Now we choose p_0 so that $E[X] \geq E[Y]/2$. For example, we can take $p_0 = \frac{1}{2}(n^{r-2}t)^{-1/(r-1)}$. Then by linearity of expectation there exists a set V_0, for which $X - Y \geq \frac{1}{4}(n/t)^{1/(r-1)}$. Fix such a set V_0 and for every edge e spanned by V_0 delete from V_0 an arbitrary vertex of e. We get an independent set U of size $|U| \geq X - Y \geq \frac{1}{4}(n/t)^{1/(r-1)}$.

The above described randomized algorithm can be easily derandomized using standard derandomization techniques (see, e.g., [2], Ch. 15). $\square$

We denote the algorithm described in Proposition 2 by $I(H, t)$. Here are its formal specifications.

Algorithm $I(H, t)$
Input: An integer t and a hypergraph $H = (V, E)$ of dimension r on n vertices, in which every $S \subset V$ of size $1 \leq |S| \leq r-1$ satisfies $|N(S)| \leq t$.
Output: An independent set U of H of size $|U| = \frac{1}{4}(n/t)^{1/(r-1)}$.

Now we are ready to give a formal description of a recursive algorithm for finding a large independent set in k-colorable hypergraphs of dimension r. Define two functions:

$$g_k(n) = \frac{1}{4} n^{\frac{1}{(r-1)(k-1)+1}}, \qquad f_k(n) = n^{1-\frac{r-1}{(r-1)(k-1)+1}} \ .$$

One can easily check that g and f satisfy

$$g_{k-1}(f_k(n)) = g_k(n), \quad \frac{1}{4}\left(\frac{n}{f_k(n)}\right)^{\frac{1}{r-1}} = g_k(n) \ .$$

Algorithm $A(H,k)$
Input: An integer $k \geq 1$ and a hypergraph $H = (V, E)$ of dimension r.
Output: A subset U of V.
$n = |V(H)|$;
if $k = 1$ take U to be an arbitrary subset of V of size $|U| = g_k(n)$ and **return**(U);
if $k \geq 2$ **then**
    while there exists a subset $S \subset V$, $1 \leq |S| \leq r-1$, such that $|N(S)| \geq f_k(n)$
        Fix one such S and fix $T \subseteq N(S)$, $|T| = f_k(n)$;
        $U = A(H[T], k-1)$;
        if U is independent in H **return**(U);
        else $H = Reduce(H, S)$;
    endwhile;
return$(I(H, f_k(n)))$;

We claim that, given a k-colorable hypergraph H as an input, the above presented algorithm finds a large independent set. This follows from the next two propositions, which can be proved by induction on k. We omit the details.

Proposition 3. *Algorithm $A(H,k)$ returns a subset of size $g_k(|V(H)|)$.*

Proposition 4. *If H is k-colorable then $A(H,k)$ outputs an independent set in H.*

Algorithm A is relatively effective for small values of the chromatic number k. Similarly to Wigderson's paper, when k is large we will switch to the following algorithm for finding an independent set. It is worth noting that the idea of partitioning the vertex set of a k-colorable hypergraph H on n into bins of size $k \log_k n$ and performing an exhaustive search for an independent set of size $\log_k n$ in each bin is due to Berger and Rompel [3].

Let

$$h_k(n) = \log_k n = \log n / \log k \ .$$

Algorithm $B(H,k)$
Input: An integer $k \geq 2$ and a hypergraph $H = (V, E)$.
Output: A subset U of $V(H)$ of size $|U| = h_k(|V(H)|)$.
$n = |V(H)|$; $h = h_k(n)$;
$l = \lfloor n/hk \rfloor$;
Partition $V(H)$ into sets $V_1, \ldots, V_l$ where $|V_1| = \ldots = |V_{l-1}| = hk$ and $hk \leq |V_l| < 2hk$;
for $i = 1$ **to** l
    for each subset U of V_i of size $|U| = h$
        if U is independent in H **then return**(U);
return an arbitrary subset of $V(H)$ of size h;

Proposition 5. *Algorithm B is correct, i.e., for a k-colorable hypergraph H on n vertices it outputs an independent set of size $h_k(n)$ in time polynomial in n.*

Proof. If H is k-colorable it contains an independent set I of size $|I| \geq n/k$. Then for some $1 \leq i \leq k$ we have $|I \cap V_i| \geq |I|/l \geq n/kl \geq h_k(n)$. Checking all subsets of V_i of size $h_k(n)$ will reveal an independent set of size $h_k(n)$. The number of subsets of size $h_k(n)$ to be checked by the algorithm does not exceed $l\binom{2h_k(n)k}{h_k(n)} = (O(1)k)^{h_k(n)} = n^{O(1)}$. $\square$

As we have already mentioned above, an algorithm for finding independent sets can be easily converted to an algorithm for coloring. The idea is very simple – as long as there are some uncolored vertices (we denote their union by W), call an algorithm for finding an independent set in the spanned subhypergraph $H[W]$, color its output by a fresh color and update W. As we have two different algorithms A and B for finding independent sets, we present two coloring algorithms C_1 and C_2, using A and B, respectively, as subroutines. Since the only difference between these two algorithms is in calling A or B, we present them jointly.

Algorithms $C_1(H,k)$ **and** $C_2(H,k)$
Input: An integer $k \geq 2$ and a hypergraph $H = (V,E)$ of dimension r.
Output: A coloring of H or a message "H is not k-colorable".

```
i = 1; W = V;
while W ≠ ∅
    for C_1(H,k): U = A(H[W],k);   for C_2(H,k): U = B(H[W],k);
    if U is not independent in H output ("H is not k-colorable") and halt;
    color U by color i; i = i + 1;
    W = W \ U;
endwhile;
return a coloring of H;
```

The correctness of both algorithms C_1 and C_2 follows immediately from that of A and B, respectively. As for the performance guarantee, it can be derived from the following easy proposition, proven, for example, implicitly in the paper of Halldórsson [11].

Proposition 6. *An iterative application of an algorithm that guarantees finding an independent set of size $f(n) = O(n^{1-\epsilon})$ in a hypergraph H on n vertices for some fixed $\epsilon > 0$, produces a coloring of H with $O(n/f(n))$ colors.*

Corollary 1.
1. *Algorithm $C_1(H,k)$ colors a k-colorable hypergraph H of dimension r on n vertices in at most $2n/g_k(n) = 8n^{1-\frac{1}{(r-1)(k-1)+1}}$ colors;*
2. *Algorithm $C_2(H,k)$ colors a k-colorable hypergraph H on n vertices in at most $2n/h_k(n) = 2n\log k/\log n$ colors.*

Proof. Follows immediately from Propositions 3, 4, 5 and 6. □

Now, given a k-colorable hypergraph H of dimension r, we can run both algorithms C_1 and C_2 and then choose the best result from their outputs. This is given by Algorithm D below.

Algorithm $D(H,k)$
Input: An integer $k \geq 2$ and a hypergraph $H = (V,E)$ of dimension r.
Output: A coloring of H or a message "H is not k-colorable".

```
Color H by Algorithm C_1(H,k);
Color H by Algorithm C_2(H,k);
if C_1 or C_2 output "H is not k-colorable", output("H is not k-colorable");
else return a coloring which uses fewer colors;
```

Until now we assumed that the chromatic number of the input hypergraph H is given in advance. Though this is not the case for general approximation algorithms, we can easily overcome this problem, for example, by trying all possible values of k from 1 to $n = |V(H)|$ and choosing a positive output of $D(H,k)$ which uses a minimal number of colors. Denote this algorithm by $E(H)$. In particular, for $k = \chi(H)$, Algorithm C_1 produces a coloring with at most $8n^{1-\frac{1}{(r-1)(k-1)+1}}$

colors, while Algorithm C_2 gives a coloring with at most $2n\log n/\log k$ colors. Hence, the approximation ratio of Algorithm E is at most

$$\min\left\{8n^{1-\frac{1}{(r-1)(k-1)+1}}/k, \frac{2n\log k}{k\log n}\right\} .$$

The first argument of the above min function is an increasing function of k, while the second one is decreasing. For $k=(1/(r-1))\log n/\log\log n$ both expressions have order $O(n(\log\log n)^2/(\log n)^2)$. Therefore we get the following result.

Theorem 2. *For every fixed $r\geq 3$, coloring of hypergraphs of dimension r on n vertices is $O(n(\log\log n)^2/(\log n)^2))$ approximable.*

4 Coloring 3-uniform 2-colorable hypergraphs

Let us start this section by defining terminology and notation to be used later. Given a 3-uniform hypergraph $H=(V,E)$, for a pair of vertices $v,u\in V$ we denote $N(u,v)=\{w\in V:(u,v,w)\in E\}$. Let also $d(u,v)=|N(u,v)|$. H is *linear* if every pair of edges of H has at most one vertex in common, that is, $d(u,v)\leq 1$ for every $u,v\in V$. Also, the *neighborhood* of $v\in V$ is defined as $N(v)=\{u\in V\setminus\{v\}:\exists w\in V,(u,v,w)\in E(H)\}$. The *independence ratio* $ir(H)$ of H is $ir(H)=\alpha(H)/|V|$.

Both papers [9] and [15] noticed that the special case of 3-uniform 2-colorable hypergraphs is different from other cases. The reason for this difference stems from the fact that for this particular case the semidefinite programming approach, pioneered for coloring problems by Karger, Motwani and Sudan [14], can be applied here as well. We refer the reader to [15] for a relevant discussion. In our algorithm we will use the semidefinite programming subroutine of [9], [15] to find a large independent set. Their analysis yields the existence of the following procedure.

Procedure *Semidef*(H)
Input: A 3-uniform 2-colorable hypergraph $H=(V,E)$ on n vertices with $m\geq n$ edges.
Output: An independent set I of size $|I|=\Omega(n^{9/8}/(m^{1/8}(\log n)^{9/8}))$.

Chen and Frieze and also Kelsen, Mahajan and Ramesh use essentially the above procedure to color a 3-uniform 2-colorable hypergraph H on n vertices in $\tilde{O}(n^{2/9})$ colors. We improve their result, again relying on the same procedure. Our main aim is to design an algorithm for finding an independent set I of size $|I|=\Omega(n^{32/41}/(\log n)^{81/82})$. As we have seen before (Prop. 6), such an algorithm can be then used to color H in $O(n^{9/41}(\log n)^{81/82})$ colors, thus giving a (slight) improvement over the result of [9], [15] in the exponent of n ($2/9=0.222\ldots$, $9/41=0.219\ldots$).

The main idea of the algorithm from [9], [15] is quite simple (given the existence of procedure *Semidef*): if there is a pair of vertices $u,v\in H$ such that $d(u,v)=\tilde{\Omega}(n^{7/9})$, then either $N(u,v)$ is independent, thus providing an independent set of a desired size, or one can apply procedure *Reduce* (see Section 3) to reduce the number of edges in H. Note that though *Reduce* creates edges of size 2, this does not cause any problem, for example, we can ignore all edges of size 2 that appear in the course of the algorithm execution and then, after having colored the hypergraph, take care of these edges. At this point we can use the fact that the graph formed by edges of size two is also 2-colorable and therefore we need at most two new colors for each color class in the coloring of the 3-uniform hypergraph. When $|E(H)| = \tilde{O}(n^{25/9})$, one can use *Semidef* to find a large independent set. We will achieve an improvement by using a more sophisticated idea in the case of dense hypergraphs. This approach is somewhat similar to that of the paper of Blum and Karger [5], where they improve the bound of Karger, Motwani and Sudan for coloring 3-colorable graphs by using more elaborate techniques for the dense case. Here is a brief outline of our algorithm. First we check whether there exists a pair of vertices $u, v \in V$ such that $d(u, v)$ is large, namely, $d_1(u, v) = \tilde{\Omega}(n^{32/41})$. If such a pair indeed exists then either $N(u, v)$ is independent or we can apply *Reduce*. Now, after all these reductions have been done, we check the number of edges of the resulting hypergraph H_1. If $|E(H_1)| = \tilde{O}(n^{113/41})$, we use *Semidef* to find an independent set of the desired size. It is intuitively clear that the more edges the hypergraph has the easier is the task of recovering its structure. We use this paradigm in the following way. If $|E(H_1)|$ is large, we find a linear subhypergraph H_2 of H_1 with all degrees at least $\tilde{\Omega}(n^{40/41})$. Now, it is rather easy to prove that there exists a vertex $v_0 \in V$ such that at least two third of the neighbors of v_0 in H_2 are colored in a color distinct from that of v in some 2-coloration of H (H_2). This implies that the neighborhood $N_{H_2}(v_0)$ has size $\tilde{\Omega}(n^{40/41})$ and spans a 2-colorable hypergraph H_3 whose vertex set contains an independent set of H of size at least $\frac{2}{3}|V(H_3)|$. Finally, we find an independent set of size $\tilde{\Omega}(|V(H_3)|^{4/5}) = \tilde{\Omega}(n^{32/41})$ in H_3 using a reminiscent of the subgraph exclusion technique applied by Boppana and Halldórsson [6].

Now we give a detailed description of the algorithm. To simplify the presentation we try to avoid the use of specific constants in some of the expressions below, hiding them in the standard O, Ω-notation, the exact values of the corresponding constants can easily be calculated. Suppose we are given a 3-uniform 2-colorable hypergraph $H = (V, E)$ on $|V| = n$ vertices. Recall that our aim is to find an independent set of size $\Omega(n^{32/41}/(\log n)^{81/82})$ in H. Let n^* be the desired size of the independent set, then

$$n^* = c\frac{n^{32/41}}{(\log n)^{81/82}}$$

for some specific constant $c > 0$. In the first phase of the algorithm we repeatedly check whether H contains a pair of vertices $u, v \in H$ such that $d(u, v) \geq n^*$. If such a pair exists, we check whether the subset $N(u, v)$ is independent in H. If it is, we return $N(u, v)$. If not, we apply procedure *Reduce* to replace all edges of size 3 passing through u, v by a single edge (u, v). In the end of this phase we get a hypergraph H_1 on n vertices in which $d(u, v) \leq n^*$ for every pair $u, v \in V$. By Prop. 1, an independent set in H_1 is also independent in H.

Now, if $|E(H_1)| = O(n^{113/41}/(\log n)^{45/41})$, we call procedure *Semidef* to find an independent set of the desired size. Otherwise, H_1 is a hypergraph with $\Omega(n^{113/41}/(\log n)^{45/41})$ edges in which there are at most n^* edges through any pair of vertices. Then H_1 contains a linear subhypergraph H_1' with the same vertex set and $\Omega(n^{81/41}/(\log n)^{9/82})$ edges. This subhypergraph can be found by a simple greedy procedure: start with $E(H_1') = \emptyset$ and as long as there exists an edge $e \in E(H_1) \setminus E(H_1')$ such that $|e \cap f| \leq 1$ for any edge $f \in E(H_1')$, add e to $E(H_1')$. If this procedure stops with $|E(H_1')| = t$ edges, then, as for every edge in $E(H_1')$ there are at most $3n^*$ other edges intersecting it in two vertices, we get $3tn^* \leq |E(H_1)| - t$, implying $t = \Omega(n^{81/41}/(\log n)^{9/82})$. Now, repeatedly delete from H_1' vertices with degree at most $t/2n$, stop when such vertices do not exist and denote the resulting subhypergraph by H_2. The total number of deleted edges does not exceed $(t/2n)n = t/2$ and therefore $E(H_2) \neq \emptyset$. Also, every vertex of H_2 has degree at least $t/2n = \Omega(n^{40/41}/(\log n)^{9/82})$. Of course, H_2 is a linear hypergraph as well.

Let us fix some 2-coloration $V(H) = C_1 \cup C_2$ of H. Also, for a vertex $v \in V(H_2)$, let $C(v) \in \{C_1, C_2\}$ denote the color class of v. Consider the sum

$$\sum_{v \in V(H_2)} \left(|N_{H_2}(v) \setminus C(v)| - 2|N_{H_2}(v) \cap C(v)|\right) =$$

$$\sum_{v \in V(H_2)} |N_{H_2} \setminus C(v)| - \sum_{v \in V(H_2)} 2|N_{H_2}(v) \cap C(v)| \ .$$

Note that every edge $e \in E(H_2)$ has exactly two vertices of one color and exactly one vertex of the opposite color. Also, since H_2 is linear, for every $v \in V(H_2)$ every vertex $u \in N(v)$ belongs to exactly one edge incident with v. Therefore every edge e contributes 4 to the first sum of the right hand side of the above equality and the same amount to the second sum. This observation shows that the sum above is equal to zero, implying in turn that at least one of the summands is non-negative. We infer that there exists a vertex $v_0 \in V(H_2)$ such that $N_{H_2}(v_0) = \Omega(n^{40/41}/(\log n)^{9/82})$ and $|N_{H_2}(v_0) \setminus C(v_0)| \geq (2/3)|N_{H_2}(v_0)|$. Let $H_3 = H[N_{H_2}(v_0)]$, $n_3 = |V(H_3)| = \Omega(n^{40/41}/(\log n)^{9/82})$. Then H_3 is 3-uniform, 2-colorable and satisfies $\alpha(H_3) \geq (2/3)n_3$. Suppose we have this vertex v_0 at hand (we can check all the vertices in polynomial time). Our aim is to find a large independent set in a hypergraph with the above properties.

At this point we apply subhypergraph exclusion techniques, similarly to the algorithm of Boppana and Halldórsson for finding a large independent set in graphs [6]. Our main tool is the following proposition.

Proposition 7. *Let $0 < a < b \leq 1$ be fixed numbers. Let G be a fixed r-uniform hypergraph with independence ratio $ir(G) = a$. Given an r-uniform hypergraph $H = (V, E)$ on n vertices with $ir(H) = b$, one can find in time polynomial in n a subset V_1 of V of size $|V_1| \geq (b - a)n$ such that the induced subhypergraph $H_1 = H[V_1]$ is G-free and $ir(H_1) \geq b$.*

Proof. Let $|V(G_0)| = n_0$. We start with $H_0 = H$ and as long as H_0 contains a copy of G we delete its vertex set from $V[H_0]$. The procedure stops when H_0 is G-free.

Denote by W the union of the vertex sets of deleted copies of G and let s be the number of deleted copies. Clearly, $s \le n/n_0$. Let $V_1 = V \setminus W$. We claim that V_1 is the desired set. In order to prove it, let I be an independent set in H of size $|I| = bn$. As $ir(G) = a$, each deleted copy of G has at most an_0 vertices in common with I. Then $|I \setminus W| \ge bn - san_0 \ge (b-a)n$, implying that $|V_1| \ge (b-a)n$. The subhypergraph $H_1 = H[V_1]$ contains the independent set $I \setminus W$, therefore

$$ir(H_1) \ge \frac{|I \setminus W|}{|V_1|} \ge \frac{bn - asn_0}{n - sn_0} \ge b$$

(recall that $a < b$). □

We will apply the above proposition several times. The first application is the following. Let G_1 be a hypergraph with vertex set $V(G_1) = \{v_1, \ldots, v_5\}$ and edge set composed of the following four edges: $\{(v_1, v_2, v_i) : 3 \le i \le 5;\ (v_3, v_4, v_5)\}$. It is easy to see that $\alpha(G_1) = 3$, implying $ir(G_1) = 3/5$. Now substituting $H = H_3$, $G = G_1$, $a = ir(G_1) = 3/5$ and $b = ir(H_3) \ge 2/3$ in Prop. 7, we find a hypergraph H_4, which is 3-uniform, 2-colorable, has $|V(H_4)| = n_4 \ge n_3/15 = \Omega(n^{40/41}/(\log n)^{9/82})$ vertices. Moreover, H_4 is G_1-free and $ir(H_4) \ge 2/3$.

Let now G_2 be defined as follows. The vertex set of G_2 consists of seven vertices $\{v_1, \ldots, v_7\}$. The edge set of G_2 has four edges: (v_1, v_2, v_5), (v_1, v_3, v_6), (v_1, v_4, v_7) and (v_5, v_6, v_7). Note that vertices v_2, v_3, v_4 have the minimal degree in G_2, therefore we will call them the *low degree vertices* of G_2. We would like to get from H_4 a G_2-free hypergraph H_5 with $\Theta(n_4)$ vertices. Though $ir(G_2) = 5/7 > 2/3$ and therefore Prop. 7 cannot be applied directly, we can observe that if I is an independent set in G_2 of size five, then I should contain all low degree vertices of G_2. Thus, when deleting copies of G_2 from H_4, we either find a large independent set in the union of the low degree vertices of the deleted copies of G_2 or end up with a G_2-free subhypergraph with many vertices. Here is a formal argument, justifying the above statement. As long as H_4 contains a copy of G_2 we delete its vertex set from $V(H_4)$. Denote the resulting subhypergraph by H_5. Let also $G^1, \ldots, G^s$ be the deleted copies of G_2, clearly, $s \le n_4/7$. Denote $W = \bigcup_{i=1}^{s} V(G^i)$, then $|W| = 7s$. Let I be an independent set in H_4 of size $|I| \ge \frac{2}{3}n_4$. Then $|I \cap W| \ge |I| + |W| - n_4 \ge 7s - n_4/3$. For $0 \le j \le 5$, define $s_j = |\{1 \le i \le s : |(V(G^i) \cap I| = j\}|$. Then

$$\sum_{j=0}^{5} s_j = s\ , \tag{1}$$

$$\sum_{j=1}^{5} js_j = |I \cap W| \ge 7s - \frac{n_4}{3}\ . \tag{2}$$

Let now $L(G^i)$ be the low degree vertices of G^i, $1 \le i \le s$, let also $L = \bigcup_{i=1}^{s} L(G^i)$. It is easy to check that if I intersects $V(G^i)$ in five vertices then

$|I \cap L(G^i)| = 3$, and also if I has four vertices in common with $V(G^i)$ then $|I \cap L(G^i)| \geq 1$. Therefore $|I \cap L| \geq 3s_5 + s_4$. Multiplying inequality (2) by 2 and subtracting (1) multiplied by 7, we get $3s_5 + s_4 \geq 7s - 2n_4/3$. This implies

$$ir(H_4[L]) \geq \frac{|I \cap L|}{|L|} \geq \frac{7s - \frac{2n_4}{3}}{3s} .$$

From here we can see that either L contains a large independent set or H_5 has $\Omega(n_4)$ vertices. Indeed, if, say, $s \geq (29/210)n_4$, then $|L| \geq (29/70)n_4$ and $ir(H_4[L]) \geq 21/29$. Then Prop. 7 can be applied to $H_4[L]$ with G being a single edge ($ir(G) = 2/3$), this way we get an independent set of size at least $(21/29 - 2/3)(29/70)n_4 = n_4/42$. Otherwise (if $s \leq (29/210)n_4$) we have $|V(H_4) \setminus W| \geq n_4/30$, implying that H_5 has $\Omega(n^{40/41}/(\log n)^{9/82})$ vertices, is 3-uniform, 2-colorable and does not contain a copy of G_1 or G_2.

Finally, let n_5 be the number of vertices of H_5. Suppose first that there exists a vertex $v \in V(H_5)$ with degree $d_{H_5}(v) \geq n_5^{8/5}/(\log n_5)^{9/5}$. Then there are two possibilities. If there exists a vertex $u \in V(H_5) \setminus \{v\}$ such that $d_{H_5}(u, v) \geq n_5^{4/5}/(\log n_5)^{9/10}$, then as H_5 is G_1-free, the subset $N_{H_5}(u, v)$ is independent in H. Otherwise, one can find $l = \Omega(n_5^{4/5}/(\log n_5)^{9/10})$ edges $e_1, \ldots, e_l$ of H_5 such that every pair of edges intersects only at v. For $1 \leq i \leq l$ choose $u_i \in e_i \setminus \{v\}$. Then, as H_5 is G_2-free, the subset $\{u_i : 1 \leq i \leq l\}$ is independent in H. If $d_{H_5}(v) = O(n_5^{8/5}/(\log n_5)^{9/5})$, we can apply procedure *Semidef* to find an independent set of size $\Omega(n_5^{4/5}/(\log n_5)^{9/10})$ in H.

Summarizing the above discussion we deduce the following corollary.

Proposition 8. *Given a 3-uniform 2-colorable hypergraph H on n vertices, one can find in time polynomial in n an independent set in H of size $\Omega(\frac{n^{32/41}}{(\log n)^{81/82}})$.*

Applying Prop. 6 we get the main result of this section.

Theorem 3. *There is a polynomial time algorithm for coloring 3-uniform 2-colorable hypergraphs on n vertices in $O(n^{9/41}(\log n)^{81/82})$ colors.*

5 Concluding remarks

Despite some progress achieved in this paper, many problems remain open and seem to be quite interesting. One of them is to develop a general approximation algorithm, aiming to match the approximation ratio $O(n(\log \log n)^2/(\log n)^3)$ of the best known graph coloring algorithm due to Halldórsson [11]. Another interesting problem is to come up with a more involved algorithm for the case of r-uniform 2-colorable hypergraphs for $r \geq 4$. Also, new ideas for the dense case of 3-uniform 2-colorable hypergraphs may lead to a further improvement in this case.

References

1. N. Alon, P. Kelsen, S. Mahajan and H. Ramesh, *Coloring 2-colorable hypergraphs with a sublinear number of colors*, Nordic J. Comput. 3 (1996), 425–439.
2. N. Alon and J. H. Spencer, **The probabilistic method**, Wiley, New York, 1992.
3. B. Berger and J. Rompel, *A better performance guarantee for approximate graph coloring*, Algorithmica 5 (1990), 459–466.
4. A. Blum, *New approximation algorithms for graph coloring*, J. ACM 31 (1994), 470–516.
5. A. Blum and D. Karger, *An $\tilde{O}(n^{3/14})$-coloring algorithm for 3-colorable graphs*, Inform. Process. Lett. 61 (1997), 49–53.
6. R. Boppana and M. M. Halldórsson, *Approximating maximum independent sets by excluding subgraphs*, Bit 32 (1992), 180–196.
7. J. Brown, *The complexity of generalized graph colorings*, Discrete Applied Math. 69 (1996), 257–270.
8. J. Brown and D. Corneil, *Graph properties and hypergraph colourings*, Discrete Math. 98 (1991), 81–93.
9. H. Chen and A. Frieze, *Coloring bipartite hypergraphs*, in: Proc. 5^{th} Intern. IPCO Conference, Lecture Notes in Comp. Sci. 1084 (1996), 345–358.
10. U. Feige and J. Kilian, *Zero knowledge and the chromatic number*, in: Proc. 11^{th} Annual IEEE Conf. on Computational Complexity, 1996.
11. M. M. Halldórsson, *A still better performance guarantee for approximate graph coloring*, Inform. Process. Lett. 45 (1993), 19–23.
12. J. Håstad, *Clique is hard to approximate within $n^{1-\epsilon}$*, in: Proc. 37^{th} IEEE FOCS, IEEE (1996), 627–636.
13. D. S. Johnson, *Worst case behaviour of graph coloring algorithms*, in: Proc. 5th S.E. Conf. on Combinatorics, Graph Theory and Computing, Congr. Numer. 10 (1974), 512–527.
14. D. Karger, R. Motwani and M. Sudan, *Approximate graph coloring by semidefinite programming*, in: Proc. 26^{th} ACM FOCS, ACM Press (1994), 2–13.
15. P. Kelsen, S. Mahajan and H. Ramesh, *Approximate hypergraph coloring*, in: Proc. 5^{th} Scandinavian Workshop on Algorithm Theory, Lecture Notes in Comp. Sci. 1097 (1996), 41–52.
16. S. Khanna, N. Linial and M. Safra, *On the hardness of approximating the chromatic number*, in: Proc. 2^{nd} Israeli Symposium on Theor. Comp. Sci., IEEE (1992), 250–260.
17. L. Lovász, *Coverings and colorings of hypergraphs*, in: Proc. 4^{th} S.E. Conf. on Combinatorics, Graph Theory and Computing, 1973, Utilitas Math., 3–12.
18. K. Phelps and V. Rödl, *On the algorithmic complexity of coloring simple hypergraphs and Steiner triple systems*, Combinatorica 4 (1984), 79–88.
19. A. Wigderson, *Improving the performance guarantee for approximate graph coloring*, J. ACM 30 (1983), 729–735.