As appearing in *BIT* **32**, 2, June 1992.

Approximating Maximum Independent Sets by Excluding Subgraphs [1]

Ravi Boppana[2] and Magnús M. Halldórsson[3]

Department of Computer Science
New York University
New York, NY 10012
U.S.A.

School of Information Science
Japan Advanced Institute of Science and Technology, Hokuriku
Tatsunokuchi, Ishikawa 920
JAPAN

Abstract

An approximation algorithm for the maximum independent set problem is given, improving the best performance guarantee known to $\mathcal{O}(n/(\log n)^2)$. We also obtain the same performance guarantee for graph coloring. The results can be combined into a surprisingly strong *simultaneous* performance guarantee for the clique and coloring problems.

The framework of *subgraph-excluding* algorithms is presented. We survey the known approximation algorithms for the independent set (clique), coloring, and vertex cover problems and show how almost all fit into that framework. We show that among subgraph-excluding algorithms, the ones presented achieve the optimal asymptotic performance guarantees.

CR Categories: F.2.2, G.2.2

Keywords: Approximation algorithms, independent sets, graph coloring.

1 Introduction

An *independent set* in a graph is a set of vertices with no edges connecting them. The problem of finding an independent set of maximum size is one of the classical $\mathcal{NP}$-hard problems. We consider polynomial time algorithms that find an independent set that is not necessarily optimal, but of a guaranteed size. The quality of the approximation is given by the ratio of the size of the maximum independent set to the size of the approximation found, and the largest such ratio over all inputs gives the *performance guarantee* of the algorithm.

A few other problems are closely related to the independent set problem. A *clique* is a set of mutually connected vertices. Since finding the maximum size clique in a graph is equivalent to finding the maximum independent set in the complement of the graph, the clique problem is for our purposes the same problem.

A *vertex cover* is a set of vertices with the property that every edge in the graph is incident to some vertex in the set. Note that vertices not in a given vertex cover must be independent, hence finding a maximum independent set is equivalent to finding a minimum vertex cover. Approximations to the two problems, however, differ widely.

The third related problem is *graph coloring*, namely finding an assignment of as few colors as possible to the vertices so that no adjacent vertices share the same color. Because the colors induce a partition of the graph into independent sets, the problems of approximating independent set and coloring are closely related. The dual problem to graph coloring is finding a *clique cover*, which is a partition of the graph into disjoint cliques.

[1] A preliminary version of this paper appeared in [9].

[2] Supported in part by National Science Foundation Grant CCR-8902522 and PYI Award CCR-9057488.

[3] Researched done at Rutgers University. Supported in part by Center for Discrete Mathematics and Theoretical Computer Science (DIMACS) fellowship.

The analysis of approximation algorithms for graph coloring started with Johnson [17] who showed that the greedy algorithm colors k-colorable graphs with $\mathcal{O}(n/\log_k n)$ colors, obtaining a performance guarantee of $\mathcal{O}(n/\log n)$. Several years later, Wigderson [21] introduced an elegant algorithm that colors k-colorable graphs with $\mathcal{O}(kn^{1/(k-1)})$ colors, which, when combined with Johnson's result, yields an $\mathcal{O}(n(\log\log n/\log n)^2)$ performance guarantee. Recently, Berger and Rompel [4] presented an algorithm that improves on Johnson's idea to obtain an $\mathcal{O}(n/(\log_k n)^2)$ coloring. When combined with Wigderson's method, they obtain an $\mathcal{O}(n(\log\log n/\log n)^3)$ performance guarantee. Halldórsson [16] improved that to $\mathcal{O}(n(\log\log n)^2/(\log n)^3)$, applying the independent set approximation algorithm of this paper. Finally, Blum has improved the best ratio for small values of k, in particular for 3-coloring from the $\mathcal{O}(\sqrt{n})$ of Wigderson and the $\mathcal{O}(\sqrt{n}/\log n)$ of Berger and Rompel, to $n^{0.4+o(1)}$ [6] and later to $n^{0.375+o(1)}$ [7].

We shall present an efficient graph coloring algorithm that colors k-colorable graphs with $\mathcal{O}(n^{(k-2)/(k-1)}/k)$ colors when $k \leq 2\log n$, and $\mathcal{O}(\log n/\log\frac{k}{\log n})$ when $k \geq 2\log n$. The algorithm strictly improves on both Johnson's and Wigderson's method.

Folklore (see [15, p. 134] attributed to Gavril) tells us that any maximal matching approximates the minimum vertex cover by a factor of two. This was slightly improved independently by Bar-Yehuda and Even [3], and Monien and Speckenmeyer [19], to a factor of $2 - \Omega(\frac{\log\log n}{\log n})$, but no further improvements have been found.

Approximating the independent set problem has seen less success. No approximation algorithm yielding a non-trivial performance guarantee has been found in the literature. One of the main results of this paper is an algorithm that obtains an $\mathcal{O}(n/(\log n)^2)$ performance guarantee for the independent set problem on general graphs, as well as several results on graphs with a high independence number.

For all these problems the optimal approximation ratios are unknown, and the gaps between the upper and lower bounds are large. The vertex cover could possibly have a polynomial time approximation schema, i.e. it could be approximable within any fixed constant greater than one. Recent results of Feige, Goldwasser, Lovász, Safra, and Szegedy [13] and Arora and Safra [2] show that the independent set problem is not approximable within a factor of $2^{\log\log n/\log\log\log n}$ unless $\mathcal{P} = \mathcal{NP}$, while results of Berman and Schnitger [5] indicate that it may not be approximable within anything less than some fixed power of n. Finally, graph coloring cannot be approximated within less than a factor of two (assuming $\mathcal{P} \neq \mathcal{NP}$) [15, p. 144], and the results of Linial and Vazirani [18] also suggest that some fixed power of n may be the best approximation that can be hoped for.

We will present lower bounds of a different kind, namely for a fixed class of algorithms, similar in spirit to the work of Chvátal [10]. We show how most approximation algorithms for the above-mentioned problems revolve around the concept of *excluding subgraphs*, and how no algorithm within that framework can do significantly better than the algorithms presented here. The techniques used have a strong connection with graph Ramsey theory, and the Ramsey-theoretic results may be of independent interest.

Graph notation

For an undirected graph $G = (V, E)$, $|G|$ is the *order* of G or the number of vertices, $\alpha(G)$ is the *independence number* of the graph or the size of the largest independent set, $i(G)$ is the *independence ratio* or the independence number divided by the order of the graph, $cl(G)$ is the *clique number*, and $\chi(G)$ is the *chromatic number* or the number of colors needed to vertex color G. For a vertex v, $N(v)$ refers to the subgraph induced by the neighbors of v and $\overline{N}(v)$ similarly the subgraph induced by the non-neighbors of v. A graph is H*-free* if it contains no subgraph (edge subset) isomorphic to H. Unless otherwise stated, G is the input graph, n is the order of G, and H is a fixed forbidden (not necessarily induced) subgraph.

2 Algorithms for Approximating Independent Sets

Suppose we decide to place a node v into a given independent set. It then suffices to search only in the non-neighborhood of v, $\overline{N}(v)$, for the remaining nodes in the set. This suggests a natural heuristic, the greedy method. We can specify its result formally as

$$\begin{array}{l} \text{Choose } v \in V(G) \\ I(G) \leftarrow \{v\} \cup I(\overline{N}(v)) \end{array}$$

This can also be formulated in a dual way for finding cliques.

$$\begin{array}{l} \text{Choose } v \in V(G) \\ C(G) \leftarrow \{v\} \cup C(N(v)) \end{array}$$

This rapid accumulation of an independent set by recursively looking at non-neighborhoods is attractive. Yet it remains disconcerting to completely ignore the *neighborhoods* of the pivot nodes, which may well contain much larger independent sets. Indeed, if we make a bad choice of a pivot node, we may be left with a minuscule set of independent vertices where there were plenty, thus Greedy performs badly in the worst case.

We are led to another rule for searching for an independent set. As before, choose a vertex and search in the non-neighborhood of that node. But this time also search in the neighborhood of the pivot node, and use whichever result is bigger. Again, a dual rule applies to the cliques. More formally,

$$\begin{array}{l} \text{Choose } v \in V(G) \\ I(G) \leftarrow \max(\{v\} \cup I(\overline{N}(v)), I(N(v))) \\ C(G) \leftarrow \max(\{v\} \cup C(N(v)), C(\overline{N}(v))) \end{array}$$

The resulting algorithm is shown in figure 2.1.

Ramsey (G)
begin
  if $G = \emptyset$ then return $(\emptyset, \emptyset)$
  choose some $v \in G$
  $(C_1, I_1) \leftarrow$ Ramsey$(N(v))$
  $(C_2, I_2) \leftarrow$ Ramsey$(\overline{N}(v))$
  return (larger of $(C_1 \cup \{v\}, C_2)$, larger of $(I_1, I_2 \cup \{v\})$)
end

Algorithm 2.1: The Ramsey Algorithm

If we look at the behavior of the algorithm, we see that it breaks the problem into a tree-like structure of subproblems. In one sense, the algorithm transforms the graph into a binary tree where each internal node is adjacent to all of its left descendants and non-adjacent to all of its right descendants. Under this interpretation, the independent set found by the algorithm is intimately related to a path in the tree with the largest number of right edges. Specifically, it consists of the leaf, and the parents of the right edges in that path. Hence, the size of the independent set found is exactly the maximum number of right edges in any path in the tree, plus one. Similarly, the size of the clique found is the maximum number of left edges in any path, plus one.

As an example, assume the input graph G contains no triangles. Clearly, the algorithm cannot find any cliques of size more than 2, and hence no path in the tree can have more than a single left edge. It follows that either the rightmost path has $\sqrt{n}$ nodes, or there are fewer than $\sqrt{n}$ paths in the tree, in which case one of them has more than $\sqrt{n}$ nodes. Either way, the algorithm finds an independent set of size no less than $\sqrt{n}$.

This formulation gives us an effective way of expressing the sizes of the approximations. A computation of the algorithm that produces a clique of size s and an independent set of size t corresponds to a binary tree where the largest number of left edges in a path is $s-1$ and the largest number of right edges is $t-1$. Let $r(s,t)$ denote the smallest integer n such that all trees of size n have paths with at least that many left or right edges. This value is one larger than the size of the largest tree with no path having $s-1$ left edges or $t-1$ right edges, which again is one less than the number of external nodes in that tree. Since each external node has an associated unique path, there can be no more than $\binom{(s-1)+(t-1)}{(t-1)}$ such nodes. Hence, $r(s,t) \leq \binom{s+t-2}{t-1}$.

The next theorem follows from the preceding discussion.

Theorem 1 *The algorithm* Ramsey *finds an independent set I and a clique C such that $r(|I|,|C|) \geq n$.*

The algorithm Ramsey is related to a classical problem in extremal graph theory. Let $R(s,t)$ denote the smallest integer n such that all graphs of order n either contain an independent set of size t or a clique of size s. This function was named after the English mathematician Frank P. Ramsey who first showed that it was well-defined. Our algorithm in his name, and the associated analysis, provides another proof to an upper bound for the Ramsey function, first proved by Erdős and Szekeres in 1934 [12].

Theorem 2 $R(s,t) \leq r(s,t) \leq \binom{s+t-2}{s-1}$

Define $t_s(n) = \min\{t \mid r(s,t) \geq n\} \geq \min\{t \mid \binom{s+t-2}{t-1} \geq n\}$. Note that if the graph contains no clique of size $k+1$, the independent set found must be of size at least $t_k(n)$. As k-colorable graphs are a subset of $(k+1)$-clique free graphs, the same bound holds for them. We can approximate $t_k(n)$ fairly accurately by $kn^{1/(k-1)}$, for $k \leq 2\log n$, and $\log n / \log \frac{k}{\log n}$, for $k \geq 2\log n$.

Notice also, that the product of $|C|$ and $t_{|C|}(n)$ is minimized when they are equal, at which point each exceeds $2\log n$. Hence,

Corollary 1 Ramsey *finds an independent set I and a clique C such that $|I| \cdot |C| \geq \frac{1}{4}(\log n)^2$.*

When carefully implemented, algorithm 2.1 involves $\mathcal{O}(n+m)$ work. Select pivot nodes according to a lexicographic-first rule. Maintain a list or array to index the vertices in the current subgraph, and divide the list into two, representing the neighborhood and non-neighborhood of the pivot node, before making the recursive call. If care is taken to conquer the smaller subproblem first, only linear (in n) extra space is required.

For small values of k, we are able to improve on Ramsey slightly. A technique by Ajtai, Komlós, and Szemerédi [1] treated by Shearer [20] as a *randomized* greedy algorithm, can be made deterministic to find an independent set in k-clique-free graphs of size $\Omega(n^{1/(k-1)}(\log n)^{(k-2)/(k-1)})$ in polynomial time.

Performance guarantee

We have seen that if the graph contains no large cliques, then Ramsey performs quite well. Unfortunately, if that precondition does not hold, we cannot make any statement about its performance. Nevertheless, if we could somehow get rid of these large cliques, we could do well on the remaining graph.

We are led to a simple method:

> Remove a maximal set of disjoint k-cliques from G, for some constant k.
> Apply Ramsey to the remaining graph.

The first concern is whether anything will be left of the graph once we have removed all vertices in disjoint k-cliques. For an arbitrary graph, the answer is no, but if the graph contains a large enough independent set, the remaining graph will be sizable. A key observation is that a clique and an independent set can share no more than a single vertex. If the independence number of the graph is at least $(1/k + \epsilon)n$ for some constant $\epsilon > 0$, then at least a fraction $\epsilon/(1 - \frac{1}{k})$ of the vertices remain.

The second problem is that finding a k-clique in the graph requires $n^{\theta(k)}$ operations for all algorithms known, hence the above algorithm is not fully polynomial in both n and k. However, we need not remove cliques that we do not run into, only those that get in our way. It suffices to remove the cliques as we go along. Recall that Ramsey finds both a clique and an independent set approximation. If the clique that Ramsey finds is small, then the independent set must be large, while if the clique is large, then we can remove it and repeat the process. This is formalized in algorithm 2.2.

CliqueRemoval (G)
begin
  $i \leftarrow 1$
  $(C_i, I_i) \leftarrow$ Ramsey (G)
  while $G \neq \emptyset$ do
    $G \leftarrow G - C_i$
    $i \leftarrow i + 1$
    $(C_i, I_i) \leftarrow$ Ramsey (G)
  od
  return $((\max_{j=1}^{i} I_j), \{C_1, C_2, \ldots, C_i\})$
end

Algorithm 2.2: Algorithm for approximating independent sets

CliqueRemoval repeatedly calls Ramsey and removes the clique found until the graph is exhausted. It then returns the largest of the independent sets found along with the sequence of cliques found. Since that collection is a partition of the vertex set into cliques, it forms an approximation to the Clique Cover problem. Moreover, if the algorithm is applied to the complement of the graph, we obtain approximations to the Clique and the Graph Coloring problems.

The following lemma is useful in relating the clique and coloring approximations (and by duality, the independent set and clique cover approximations).

Lemma 1 *Let* A *be an algorithm that guarantees finding independent sets of size $f(n)$ in k-colorable graphs of order n, where f is a positive, non-decreasing function. Then an iterative application of* A *on a k-colorable graph G produces a coloring of G with no more than $\sum_{i=1}^{n} 1/f(i)$ colors.*

We can now prove tight bounds on the sizes of the approximations.

Theorem 3 *Given a graph G, let k be the smallest integer such that $\alpha(G) > n/k$, and let $\epsilon = \alpha(G)/n - 1/k$. Define $t_s(n)$ as before.*

The algorithm CliqueRemoval *finds an independent set approximation I, and a clique cover approximation CC, such that*

$$|I| \geq \max(t_k(\epsilon n), t_{k+1}(\tfrac{n}{k^2})) \quad \textit{and} \quad |CC| \leq \tfrac{5n}{(\log n)^2}|I|$$

Proof. Let us first consider the first claim. Since the independence fraction of the graph is strictly greater than $1/k$, the algorithm must eventually find no k-clique, at which point

$(\epsilon/(1-1/k))n \geq \epsilon n$ vertices remain in the graph. Also, the point when the algorithm finds no $(k+1)$-clique occurs even earlier, when at least $(\epsilon + 1/k - 1/(k+1)) \cdot (1/(1-1/k)) \cdot n \geq n/k^2$ vertices remain in the graph. The bound then follows from theorem 1.

For the second claim, we shall, for pure convenience, analyze the approximations guaranteed for the Clique and Coloring problems, with the understanding that the same applies immediately to the Independent Set and Clique Cover problems, respectively. Let C be the clique approximation, and $\{I_1, \ldots, I_{\text{Colors}}\}$ be the coloring approximation.

Recall that the approximations produced by Ramsey satisfy $|C_i| \cdot |I_i| \geq (\frac{1}{2}\log|G_i|)^2$. Hence, if $f(m)$ represents the value of $|I_i|$ when $|G_i| = m$, then $f(m) \geq \frac{1}{4}(\log m)^2/|C_i| \geq \frac{1}{4}(\log m)^2/|C|$. Applying lemma 1, we get that $\textsc{Colors} \leq \sum_{i=1}^{n} 4|C|/(\log i)^2 \leq 5n|C|/(\log n)^2$. ∎

Now consider the product of the two performance guarantees.

$$\frac{cl(G)}{|C|}\frac{\textsc{Colors}}{\chi(G)} \leq \frac{5n}{(\log n)^2}\frac{cl(G)}{\chi(G)}$$

Since $cl(G)$ is never greater than $\chi(G)$, this bound immediately yields the claimed $\mathcal{O}(n/(\log n)^2)$ individual bounds on the performance guarantees. For classes of instances for which the measures are apart, the performance guarantees are even stronger. In particular, random graphs almost always have a clique number asymptotically $2\log n$ and chromatic number $n/(2\log n)$, and for graphs with these parameters the product of the performance guarantees is a constant (no more than 20). This also implies that the stated relationship between the sizes of the two approximations is optimal within a constant. [4]

The above approximation and performance guarantee for the independent set (and by duality the clique) problem are the best known. The approximation for graph coloring is also the best known for graphs with chromatic number between $\sqrt{\frac{\log n}{\log\log n}}$ and $\frac{\log n}{\log\log n}$. For graphs with a smaller chromatic number the method of Blum [7] performs best, while for larger chromatic numbers Halldórsson's [16] improvement of Berger and Rompel's result [4] is stronger.

3 Subgraph-Excluding Algorithms

Let us formally define a framework that properly captures all the algorithms for finding independent sets given in this paper.

Definition 1 *An algorithm* A*, with an associated fixed graph H and function f, is a* Ramsey-type *algorithm if, for every H-free graph G, it guarantees finding an independent set of size at least $f(|G|)$.*

Definition 2 *An algorithm* B_H *is a* subgraph-exclusion *algorithm if, given arbitrary graph G, it is of the form:*

1. *Ensure that G contains no copy of the subgraph H, and*
2. *Apply a Ramsey-type algorithm on G.*

There are a few ways in which such an algorithm can exclude a subgraph H:

Remove: All copies of the forbidden subgraph, or parts of it, can be pulled out of the graph sequentially. A necessary and sufficient precondition for the removal process to retain at least a constant fraction of the vertices is that $i(H) < i(G) + \epsilon$, for some constant $\epsilon > 0$.

Forbid: The exclusion of the subgraph can be built into the statement of the problem. This applies particularly to the graph coloring problem. For instance, the clique on $k+1$ vertices cannot appear in k-colorable graphs.

[4] When read with an algorithmic frame of mind, this relationship and the resulting performance guarantees can be found in a 1967 paper of Erdős [11].

Merge: In certain cases, vertices can be fused together, causing a certain type of a subgraph to disappear.

The issue becomes finding graphs H that force graphs free of H to contain large independent sets, as well as coming up with algorithms to actually find those independent sets in H-free graphs. The previous section described algorithms that use cliques. Other subgraphs discussed in this section include odd cycles, wheels, and color-critical subgraphs. The following section will then illustrate that these subgraphs are in some sense the best of their kind.

Wheels

A *wheel*, denoted by $W_{p,m}$, is a graph that consists of an odd cycle of $m \geq 3$ nodes, and $p \geq 0$ *spokes*, which are nodes that connect to all other nodes in the graph. A wheel with p spokes is referred to as a p-wheel. The clique number of an p-wheel is $p+2$ (except when $m = 3$), whereas the chromatic number is $p+3$.

Note that if a graph does not contain a p-wheel, then no neighborhood graph can contain a $(p-1)$-wheel nor can any non-neighborhood graph contain a p-wheel. Hence we obtain the same recursive relationship as in the Ramsey algorithm. Only the base case is different; we capitalize on the fact that coloring a bipartite graph is easily solvable in linear time.

> **WheelFreeRamsey** (G)
> begin
>   if (G is bipartite) then return (some edge, the larger color set)
>   choose some $v \in G$
> $(W_1, I_1) \leftarrow$ WheelFreeRamsey($N(v)$)
> $(W_2, I_2) \leftarrow$ WheelFreeRamsey($\overline{N}(v)$)
>   return (larger of ($W_1 \cup \{v\}, W_2$), larger of ($I_1, I_2 \cup \{v\}$))
> end

Algorithm 3.1: Ramsey Algorithm for Wheels

Define $R(W_p, K_t)$ to be the minimal n such that all graphs of order n contain some p-wheel or an independent set of size t. We find that $R(W_p, K_t) \leq R(W_{p-1}, K_t) + R(W_p, K_{t-1})$ and $R(W_0, K_t) = 2t - 1$ and $R(W_p, K_2) = p + 3$. An inductive argument shows that $R(W_p, K_t) \leq 2\binom{p+t}{t-1}$, only a factor of two from the upper bound of the regular Ramsey function.

Given a graph with no $(k-2)$-wheels, WheelFreeRamsey finds an independent set of size at least $\Omega(kn^{1/(k-1)})$. By applying a version of algorithm 2.2 that utilizes WheelFreeRamsey, we can color a graph without $(k-2)$-wheels using $\mathcal{O}(n^{(k-2)/(k-1)}/k)$ colors.

Algorithm 3.1 is closely related to Wigderson's coloring algorithm [21]. By considering the whole uncolored portion of the graph in each iteration, instead of fully coloring the pivot nodes' neighborhoods before coloring their non-neighbors, WheelFreeRamsey improves the approximation by a factor of k. Also, by focusing alternately on neighborhoods and non-neighborhoods it gains another factor of k. Wigderson's method, however, has the advantage of $\mathcal{O}(\chi(G)\ (n+m))$ time complexity, compared to the $\mathcal{O}(\text{Colors}\ (n+m))$ complexity of our method. Compared with the graph coloring algorithm deduced from the Ramsey algorithm for clique-free graphs, this algorithm improves the exponent from $\frac{k-1}{k}$ to $\frac{k-2}{k-1}$.

Short odd cycles

For graphs with independence ratio in the range of $(\frac{1}{3} + \epsilon, \frac{1}{2})$, the Ramsey algorithm obtains an independent set approximation of $\Omega(\sqrt{n})$ by removing triangles. Families of odd cycles as excluded subgraphs allow us to refine the approximations in this range.

The method starts by removing all odd cycles of size up to $2k+1$. In contrast to the case for cliques, this can be done in linear time independent of k. Note that a cycle of length $2k+1$

has an independence ratio $\frac{k}{2k+1}$. So if $i(G) > \frac{k}{2k+1}$, we can remove these cycles and then apply algorithm 3.2.

```
OddCycleFreeApproximation (G, k)
 { Graph G contains no odd cycles of length 2k + 1 or shorter}
begin
  while G ≠ ∅ do
    choose any vertex v in V(G).
    V_i ← vertices of distance i from v.
    S_i ← V_i ∪ V_{i-2} ∪ ...
    Determine i such that |S_{i+1}| ≤ n^{1/(k+1)} |S_i|.
    I ← I ∪ S_i
    G ← G − S_i − S_{i+1}
  od
  return I
end
```

Algorithm 3.2: Algorithm for independent sets on graphs with no short odd cycles

Since each independent set S_i selected causes only $n^{1/(k+1)}$ times as many other nodes to be removed from the graph, the graph is not exhausted until an independent set of at least $n^{k/(k+1)}$ has been collected. Assume there was no i satisfying $|S_{i+1}| \leq n^{1/(k+1)}|S_i|$. Then $|S_k| > n^{1/(k+1)}|S_{k-1}| > n^{2/(k+1)}|S_{k-2}| > \cdots > n^{k/(k+1)}|S_0| = n^{k/(k+1)}$, and the problem is solved.

Since each vertex and each edge are looked at only once, the algorithm runs in linear time. On the other hand, when applied to general graphs the algorithm must be run for many different values of k, in which case it may be useful to combine the cycle removal process (see [19]).

The technique of Ajtai, Komlós, and Szemerédi can also be applied here. When k is fixed, we can find an independent set of size $\Omega(n^{k/(k+1)}(\log n)^{1/(k+1)})$ in polynomial time for graphs with no odd cycles of length $2k+1$ or less.

Color-critical graphs

For graphs of fixed chromatic number, an algorithm A. Blum [6, 7] improves on the previously mentioned algorithm of Wigderson. In particular, it uses only $n^{3/8+o(1)}$ colors for 3-colorable graphs, down from $\mathcal{O}(\sqrt{n})$. His complicated method can be summarized in the following three steps:

1. Destroy all copies of the subgraphs $K_4 - e$ and 1-2-3 graphs by collapsing certain pairs of nodes.
2. Classify vertices according to degree, producing a polynomial number of subgraphs, one of which has an independence ratio close to one half.
3. Apply algorithm 3.2 on each of these subgraphs.

The graph $K_4 - e$ is the clique on 4 vertices with one edge removed. A "1-2-3 graph" is our term for a graph with three specific parts: A, consisting of two disconnected nodes; B, an independent set of at least 3 nodes; and C, an odd cycle, where parts A and C are completely disconnected, A and B are completely connected, and the connections between B and C are such that each node in C is connected to some node in B. Since C requires three colors, B needs two, and thus the two nodes in A must have the same color under any legal 3-coloring of the subgraph, whence the name 1-2-3. Similarly, the two disjoint nodes in $K_4 - e$ must share the same color.

The first and the third steps are strictly Ramsey-type, whereas the second does use the size of the independent sets promised by the k-colorability property. Hence the algorithm appears to lack the "forgetfulness" property of Ramsey-style algorithms.

4 Limitation results

The main result of this section is that excluding subgraphs other than cliques and series of odd cycles does not help much in forcing a graph to contain a large independent set. This implies that no subgraph removal algorithms, even super-polynomial ones, can yield asymptotically better performance guarantees for the maximum independent set, graph coloring, and vertex cover problem than the algorithms given.

Let us extend the Ramsey function from cliques to arbitrary graphs. Let $R(H, K_t)$ denote the minimal n such that every graph on n vertices either contains a copy of the graph H or has an independent set of size t. Note that H does not need to be isomorphic to a *vertex induced* subgraph of G, only that all the edges of H be contained in such a subgraph. It immediately follows that $R(H, K_t) \geq R(H', K_t)$ whenever H' is an edge-subset of H. Obtaining an upper bound on $R(H, K_t)$ shows that not all H-free graphs contain very large independent sets, showing a limitation on the power of excluding H.

A few definitions are in order. For a graph H, let $e(H)$ be the number of edges, and $\rho(H)$ denote the maximum of $e(H')/|H'|$ over all subgraphs H' of H. Extend these definitions to a collection $\mathcal{H}$ of graphs. Define $i(\mathcal{H})$ to be the maximum of $i(H)$ over all H in $\mathcal{H}$. Define $\rho(\mathcal{H})$ and $\chi(\mathcal{H})$ to be the minimum of $\rho(H)$ and $\chi(H)$, respectively, over all H in $\mathcal{H}$. Also, $R(\mathcal{H}, K_t)$ is the minimal n such that every graph on n vertices either contains a copy of some H in $\mathcal{H}$ or has an independent set of size t.

There are some well-known relations between these quantities. One relation is $\chi(H)i(H) \geq 1$, which holds because a coloring is just a partition into independent sets. Another relation is $\chi(H) \leq 2\rho(H) + 1$, which holds because H has a vertex of degree $2\rho(H)$ or less. Both relations generalize to a collection of graphs.

We will give the central theorem for a function slightly stronger than ρ.

Define $\rho'(H) = \min \frac{e(H')-1}{|H'|-2}$ where H' ranges over subgraphs of H on at least 3 vertices. Similarly extend ρ' to a collection of graphs $\mathcal{H}$. The value of ρ' is always at least as large as ρ, and for small graphs the improvement makes a difference.

Theorem 4 $R(\mathcal{H}, K_t) = \Omega((\frac{t}{\log t})^{\rho'(\mathcal{H})})$

Proof. The proof follows the probabilistic method using the Lovász local lemma. We follow closely the presentation of Bollobás [8, p.287] of a lower bound on the ordinary Ramsey numbers. We give a proof only for a singleton collection $\mathcal{H} = \{H\}$; the general case is similar.

Let $l = e(H)$ and $s = v(H)$, and let $r = \frac{l-1}{s-2}$. We claim that $R(H, K_t) \geq c(\frac{t}{\log t})^r$, for $c = 1/(4(4r)^r \binom{s}{2}^{1/(s-2)})$.

Find positive numbers a, b, and ϵ such that $0 < \epsilon < 1/4$, $a > 2(r + b - 1)$, and $b > (1+\epsilon)^2 a^l c^{s-2} \binom{s}{2}$. Such a choice is possible since if we take $\epsilon = 0$, and replace the inequalities above by equalities, the solutions for a and b are positive.

Consider G in $\mathcal{G}(n,p)$, a random graph on n vertices with edge probability p, with $n = c(t/\log t)^r$ and $p = a \log t/t$. Let U be the space of all vertex subsets of size s, and W the space of all t-sets. Let A_S be the event that a given instance S of U contains the forbidden subgraph H, and let B_T be the event that a given instance T of W is independent. The A_S's all have the same probability, which we denote by p_A, and similarly p_B denotes the probability of each B_T.

Consider the graph on $U \cup W$ in which two vertices are joined by an edge iff the corresponding subsets of V have at least two vertices in common. This is precisely the graph of dependencies among the events $\{A_i : i \in U\} \cup \{B_j : j \in W\}$. Let d_A be the number of events in U intersecting with A_S, d_B the number of events in W intersecting with B_T, d_{AB} the number of events in W intersecting with A_S, and d_{BA} the number of events in W intersecting with B_T.

We have that

$$p_A \leq p^l s! = (\frac{a \log t}{t})^l s!$$

$$p_B = (1-p)^{\binom{t}{2}} \leq e^{-p\binom{t}{2}} = t^{-a(t-1)/2}$$

$$d_A \leq \binom{s}{2}\binom{n}{s-2} \leq 3n^{s-2} \leq 3c^{s-2}(\frac{t}{\log t})^{l-1}$$

$$d_B \leq \binom{n}{t} < (\frac{en}{t})^t$$

$$d_{AB} \leq \binom{n}{t} < (\frac{en}{t})^t$$

$$d_{BA} \leq \binom{t}{2}\binom{n}{s-2} < \frac{t^2}{2(s-2)!} n^{s-2} = \frac{t^2}{2(s-2)!} c^{s-2} (\frac{t}{\log t})^{l-1}$$

Bollobás derived the following version of a theorem of Lovász, for dependence graphs with two kinds of events.

Fact 1 *If there are δ_A, δ_B such that $1 < \delta_A < 1/(2p_A)$, $1 < \delta_B < 1/(2p_B)$,*

$$\log \delta_A \geq (1 + \delta_A p_A)(d_A p_A)\delta_A + (1 + \delta_B p_B)(d_{AB} p_B)\delta_B$$

and

$$\log \delta_B \geq (1 + \delta_A p_A)(d_{BA} p_A)\delta_A + (1 + \delta_B p_B)(d_B p_B)\delta_B$$

then

$$Pr\left[\bigcap_{i\in U} \overline{A}_i \cap \bigcap_{j \in W} \overline{B}_j\right] > 0$$

Our main claim will follow if we can find values for δ_A and δ_B that satisfy the conditions in the above fact. We claim that $\delta_A = 1 + \epsilon$ and $\delta_B = t^{bt}$ will suffice, provided t is sufficiently large. We find that

$$d_A p_A \leq 3c^{s-2}(\frac{t}{\log t})^{l-1} \cdot (\frac{a \log t}{t})^l s! = \mathcal{O}(\frac{\log t}{t}) = o(1)$$

and

$$d_{AB} p_B \delta_B \leq (\frac{en}{t})^t \cdot t^{-a(t-1)/2} \cdot t^{bt} = t^{t((l-1)/(s-2) - a/2 + b - 1) + o(t)} = o(1)$$

since by the bound on a, the exponent of t is negative if t is large enough. Hence, the first condition of the fact holds.

By the last inequality, $d_B p_B \delta_B = o(1)$. Furthermore,

$$\begin{aligned}\delta_A p_A d_{BA} &= (1+\epsilon)[a(\log t)/t]^l s! \frac{t^2}{2(s-2)!} c^{s-2} (t/(\log t))^{l-1} \\ &= (1+\epsilon) a^l c^{s-2} \binom{s}{2} (t \log t) \quad < \quad (1+\epsilon)^{-1} \log \delta_B\end{aligned}$$

because of the constraint on b. Since $\delta_A p_A = o(1)$ this implies the second condition, if t is sufficiently large.

We have shown that $\Pr[G$ contains an H or $\overline{K_t}] < 1$ and thus there exists a graph on n vertices that contains no independent set of size t, nor a subgraph isomorphic to H. Hence, the Ramsey number $R(H, K_t)$ must be larger than $n = c(t/\log t)^r$.

Finally, since the above argument applies as well to any subgraph of H, in particular, H' such that $\frac{e(H')-1}{v(H')-2} = \rho'(H)$, the fact that $R(H, K_t) \geq R(H', K_t)$ allows us to improve the exponent in the value of n from $r = \frac{e(H)-1}{v(H)-2}$ to $\rho'(H) = \max_{H' \in H} \frac{e(H')-1}{v(H')-2}$. ∎

Recall that Blum's algorithm made use of subgraphs that contain two nodes that must be of the same color under any legal 3-coloring. A graph is k-*avoidable* iff it has a pair of vertices that get assigned the same color for every k-coloring of the graph. Note that this is vacuously true for non-k-colorable graphs. Alternatively, k-avoidable graphs can be characterized as being no more than one edge away from being $(k+1)$-chromatic. A collection $\mathcal{H}$ is k-avoidable iff every H in $\mathcal{H}$ is.

Corollary 2 *For every positive integer k, if $\mathcal{H}$ is k-avoidable, then $R(\mathcal{H}, K_t) = \Omega((\frac{t}{\log t})^{k/2})$.*

Proof. If $H \in \mathcal{H}$ is k-avoidable, then $H + e$ is $(k+1)$-chromatic for some edge e. Hence $\rho(H+e) \geq \frac{k}{2}$. But $\rho'(H) \geq \min_{H' \in H+e} \frac{e(H')-2}{|H'|-2} \geq \rho(H+e)$, when $\rho(H+e) \geq 1$. This holds for all H in $\mathcal{H}$, hence the conclusion follows from theorem 4. ∎

This result implies that a Ramsey-type algorithm on a k-colorable graph that relies solely on the lack of some set of k-avoidable subgraphs cannot guarantee finding an independent set of size more than $\mathcal{O}(n^{2/k} \log n)$, and hence cannot guarantee a coloring with less than $\Omega(n^{1-2/k}/\log n)$ colors. As an example, no such algorithm can guarantee coloring a 3-colorable graph with less than $\Omega(n^{1/3}/\log n)$ colors.

We can make a stronger statement regarding the 3-coloring problem.

Theorem 5 *If $\mathcal{H}$ is 3-avoidable, then $\rho'(\mathcal{H}) \geq \frac{3}{2} + \frac{1}{26}$.*

Proof. Let H be a 3-avoidable graph in $\mathcal{H}$, and $H + e$ be 4-chromatic. A *4-critical* graph is a 4-chromatic graph with the property that removing any node will make it 3-colorable. Gallai [14] showed that 4-critical graphs, with the exception of K_4, have an edge-to-vertex ratio of at least $\frac{3}{2} + \frac{1}{26}$. If $H + e$ contains a K_4, then $\rho'(H) \geq \rho'(K_4 - e) = \frac{5-1}{4-2} = 2$. Otherwise, $\rho(H+e) \geq \frac{e(H^*)}{|H^*|} \geq \frac{3}{2} + \frac{1}{26}$, by Gallai's result. In either case, $\rho'(H) \geq \frac{3}{2} + \frac{1}{26}$ for any H in $\mathcal{H}$. ∎

As a result, Ramsey-type algorithms require at least $\Omega(n^{1-1/(\frac{3}{2}+\frac{1}{26})}/\log n) = \Omega(n^{.35}/\log n)$ colors on 3-colorable graphs. Notice that Blum's technique also breaks down in the region of $n^{1/3}$ [7], even though it is not known to be of a subgraph-excluding type.

Let us now derive a limitation for general graphs. It can be shown, in the spirit of the bounds on the diagonal Ramsey function $R(s, s)$, that if $\mathcal{H}$ is a t-avoidable collection then $R(\mathcal{H}, K_t) = 2^{\Omega(t)}$. Hence if all graphs of order n contain either a subgraph H in the t-chromatic collection $\mathcal{H}$ or an independent set of size t, then t must be $\mathcal{O}(\log n)$. Hence no Ramsey-type algorithm that relies solely on the lack of avoidable subgraphs can obtain a better performance guarantee than $\Omega(n/(\log n)^2)$ for graph coloring.

Our emphasis so far on graph coloring is because the lower bounds for graph coloring are also lower bounds for the independent set problem. Since $i(H) < \frac{1}{k}$ implies that $\chi(H) \geq k+1$, corollary 2 holds as well for graphs with large independence ratio. Similarly, the limitation result on performance guarantees for the general coloring problem carries over immediately to the maximum independent set problem.

Limitation results for odd cycles

Our next goal is to show that our cycle-based algorithm is close to optimal for graphs with independence ratio near $\frac{1}{2}$.

We need the following structural result on graphs without short odd cycles.

Theorem 6 *For every positive integer k, if $\rho(H) \leq 1 + \frac{1}{4k+2}$, and H contains no odd cycles of length $2k-1$ or less, then $i(H) \geq \frac{k}{2k+1}$.*

Proof. By induction on the number of vertices in H. If there is a vertex v of degree 0 or 1, then remove v and its neighbor from the graph. By induction, the remaining graph has independence ratio at least $\frac{k}{2k+1}$. But adding v to the largest independent set of the remaining graph shows that H itself has independence ratio greater than $\frac{k}{2k+1}$.

Thus, assume that every vertex has degree 2 or more. Suppose there is a cycle passing through only vertices of degree exactly 2. Since H has no odd cycles of length $2k-1$ or less, the independence ratio of this cycle is at least $\frac{k}{2k+1}$. Then we could apply induction to the remainder of the graph and be finished.

Thus, assume there are no such cycles. Let H' be the subgraph induced by the vertices of degree exactly 2. The subgraph H' must be the disjoint union of paths, so $i(H') \geq \frac{1}{2}$. Since $\rho(H) \leq 1 + \frac{1}{4k+2}$, the subgraph H' contains at least a fraction $1 - \frac{1}{2k+1}$ of all the vertices. Therefore $i(H)$ is at least $\frac{1}{2}(1 - \frac{1}{2k+1}) = \frac{k}{2k+1}$, which completes the proof. ∎

Finally, we can prove the following limitation result.

Corollary 3 *For every positive integer k, if $i(\mathcal{H}) < \frac{k}{2k+1}$, then $R(\mathcal{H}, K_t) = \Omega(\frac{t}{\log t})^{1+1/(4k+2)})$.*

Proof. By theorem 6 above, every H in $\mathcal{H}$ either has an odd cycle of length $2k-1$ or less, or satisfies $\rho(H) \geq 1 + \frac{1}{4k+2}$. The first case implies that $\rho'(H) \geq \frac{2k-2}{2k-3} = 1 + \frac{1}{2k-3}$, thus in either case $\rho'(H) \geq 1 + \frac{1}{4k+2}$. ∎

This result implies that for a graph with independence ratio $\frac{k}{2k+1}$, no subgraph-removal algorithm can guarantee an independent set larger than $\mathcal{O}(n^{1-1/(4k+3)})$. Recall that our cycle-based algorithm will find an independent set of size $\Omega(n^{1-1/k})$ and thus the cycle-based algorithm is close to optimal.

The above result also implies that for approximately solving the vertex cover problem, no subgraph-removal algorithm can achieve a performance guarantee better than $2 - \theta(\frac{\log \log n}{\log n})$, the performance guarantee obtained by the algorithm of Monien and Speckenmeyer.

5 Discussion

The central open problem is determining the best possible performance guarantees for the independent set and graph coloring problems. All signs seem to indicate that the bulk of the improvements must come from the lower bounds. While general lower bounds are hard to come by, we would like to see lower bounds for further classes of algorithms or models of computation.

References

[1] M. Ajtai, J. Komlós, and E. Szemerédi. A note on Ramsey numbers. *J. Combin. Theory Ser. A*, 29:354–360, 1980.

[2] S. Arora and S. Safra. Probabilistic checking of proofs; a new characterization of NP. In *Proc. 33nd Ann. IEEE Symp. on Found. of Comp. Sci.*, pages 2–13, Oct. 1992.

[3] R. Bar-Yehuda and S. Even. A $2 - \frac{\log \log n}{2 \log n}$ performance ratio for the weighted vertex cover problem. Technical Report #260, Technion, Haifa, Jan. 1983.

[4] B. Berger and J. Rompel. A better performance guarantee for approximate graph coloring. *Algorithmica*, 5(4):459–466, 1990.

[5] P. Berman and G. Schnitger. On the complexity of approximating the independent set problem. In *Proc. Symp. Theoret. Aspects of Comp. Sci. Lecture Notes in Comp. Sci. # 349*, pages 256–268. Springer-Verlag, 1989.

[6] A. Blum. An $\tilde{O}(n^{.4})$ approximation algorithm for 3-coloring. In *Proc. 21st Ann. ACM Symp. on Theory of Computing*, pages 535–542, 1989.

[7] A. Blum. Some tools for approximate 3-coloring. In *Proc. 31st Ann. IEEE Symp. on Found. of Comp. Sci.*, pages 554–562, Oct. 1990.

[8] B. Bollobás. *Random Graphs*. Academic Press, 1985.

[9] R. B. Boppana and M. M. Halldórsson. Approximating maximum independent sets by excluding subgraphs. In *Proc. of 2nd Scand. Workshop on Algorithm Theory. Lecture Notes in Computer Science #447*, pages 13–25. Springer-Verlag, July 1990.

[10] V. Chvátal. Determining the stability number of a graph. *SIAM J. Comput.*, 6(4), Dec. 1977.

[11] P. Erdős. Some remarks on chromatic graphs. *Colloq. Math.*, 16:253–256, 1967.

[12] P. Erdős and G. Szekeres. A combinatorial problem in geometry. *Compositio Math.*, 2:463–470, 1935.

[13] U. Feige, S. Goldwasser, L. Lovász, S. Safra, and M. Szegedy. Approximating clique is almost NP-complete. In *Proc. 32nd Ann. IEEE Symp. on Found. of Comp. Sci.*, Oct. 1991. To appear.

[14] T. Gallai. Kritische graphen I. *Publ. Math. Inst. Hungar. Acad. Sci.*, 8:165–192, 1963. (See Bollobás, B. *Extremal Graph Theory.*, Academic Press, 1978, page 285).

[15] M. R. Garey and D. S. Johnson. *Computers and Intractibility: A Guide to the Theory of NP-completeness*. Freeman, 1979.

[16] M. M. Halldórsson. A still better performance guarantee for approximate graph coloring. Technical Report 90–44, DIMACS, June 1990.

[17] D. S. Johnson. Worst case behaviour of graph coloring algorithms. In *Proc. 5th Southeastern Conf. on Combinatorics, Graph Theory, and Computing. Congressus Numerantium X*, pages 513–527, 1974.

[18] N. Linial and V. Vazirani. Graph products and chromatic numbers. In *Proc. 30th Ann. IEEE Symp. on Found. of Comp. Sci.*, pages 124–128, 1989.

[19] B. Monien and E. Speckenmeyer. Ramsey numbers and an approximation algorithm for the vertex cover problem. *Acta Inf.*, 22:115–123, 1985.

[20] J. B. Shearer. A note on the independence number of triangle-free graphs. *Discrete Math.*, 46:83–87, 1983.

[21] A. Wigderson. Improving the performance guarantee for approximate graph coloring. *J. ACM*, 30(4):729–735, 1983.