Nordic Journal of Computing 1(1994), 475–492.

IMPROVED APPROXIMATIONS OF INDEPENDENT SETS IN BOUNDED-DEGREE GRAPHS VIA SUBGRAPH REMOVAL

MAGNÚS M. HALLDÓRSSON
School of Information Science
Japan Advanced Institute of Science and Technology
Tatsunokuchi, Ishikawa 923-12, Japan.
magnus@jaist.ac.jp

JAIKUMAR RADHAKRISHNAN
Theoretical Computer Science Group
Tata Institute of Fundamental Research
Bombay, India 400 005.
jaikumar@tifrvax.bitnet

Abstract. Finding maximum independent sets in graphs with bounded maximum degree Δ is a well-studied NP-complete problem. We introduce an algorithm schema for improving the approximation of algorithms for this problem, which is based on preprocessing the input by removing cliques.

We give an implementation of a theorem on the independence number of clique-free graphs, and use it to obtain an $O(\Delta/\log\log\Delta)$ performance ratio with our schema. This is the first $o(\Delta)$ ratio for the independent set problem. We also obtain an efficient method with a $\Delta/6(1+o(1))$ performance ratio, improving on the best performance ratio known for intermediate values of Δ.

CR Classification: F.2.2, G.2.2

Key words: analysis of algorithms, approximation algorithms, independent sets

1. Introduction

An *independent set* in a graph is a set of vertices in which no two are adjacent. The problem of finding an independent set of maximum cardinality is of central importance in graph theory and combinatorial optimization. It is however NP-hard; thus no efficient algorithms can be expected.

A promising approach for dealing with intractability is in developing heuristics that find quality approximate solutions. The *performance ratio* of such an algorithm is defined to be the worst-case ratio of the size of the optimal solution to the size of the algorithm's solution. However, in spite of considerable effort, no algorithm is known for the independent set problem with a performance ratio less than $\theta(n/\log^2 n)$ [9], where n is the number of vertices in the input graph. Results in recent years on interactive proof

Received May 1994. Accepted September 1994.

systems, culminating in the celebrated paper of Arora et al. [5], show that no constant-factor approximation can be expected, and in fact, an $n^{1/4}$ ratio appears out of reach [6].

Given this apparent hardness of the general problem, it is natural to ask what restrictions on the input lead to better approximations. Perhaps the most natural and frequently occurring case is when the maximum vertex degree is bounded above by a constant. Just as the independent set (or clique) problem occurs in various contexts when modeling pairwise conflicts among elements, the bounded-degree variant occurs naturally when key parameters are fixed.

The independent set problem remains NP-complete on graphs of bounded maximum degree Δ, but approximation becomes considerably easier. In fact, any algorithm that finds a *maximal* independent set has a performance ratio of Δ. On the other hand, for any fixed $\Delta \geq 3$, there is a fixed constant $\epsilon > 1$ for which $1+\epsilon$-approximation becomes NP-hard [5, 20]. These hardness results can be amplified via so-called randomized graph products, which have been derandomized by Alon et al. [3], and by additional analysis it can be shown that there is a $\epsilon > 0$ for which $O(\Delta^\epsilon)$ approximation is NP-hard [12]. This naturally brings up the issue of the exact approximability of the problem; the current paper narrows the gap somewhat.

We present an algorithm schema that involves removing small cliques from the graph. The idea – which originates in [9], with traces back to Erdős [11] – is rooted in results of graph theory that state that graphs without small cliques as induced subgraphs must contain larger independent sets than do general graphs, and moreover, these larger solutions can be found effectively. For graphs with few disjoint cliques, we can manually remove all the cliques and find the promised improved solution on the remainder. On the other hand, graphs with many disjoint cliques cannot contain very large independent sets, providing an upper bound on the optimal solution. In either case, our performance ratio will be improved.

We use this schema to give a positive answer to a tantalizing question: Given that all approximation algorithms for the bounded-degree independent set problem have so far merely improved the coefficient in front of Δ, is a $o(\Delta)$ performance ratio possible? We present an algorithm with an $O(\Delta/\log\log\Delta)$ performance ratio. An essential component is a deterministic implementation of an existential theorem of Ajtai, Erdős, Komlós, and Szemerédi [1] on the independence number of sparse graphs containing no small cliques.

We also analyze the schema in combination with two practical algorithms: a simple local search algorithm of Khanna, Motwani, Sudan and Vazirani [17], and an algorithm of Shearer [22] for triangle-free graphs. We obtain a $\Delta/6(1+o(1))$ ratio for this combination that improves on previous results for intermediate to large values of Δ.

The results reported here previously appeared in abstracted form as a part of [15], and to a lesser extent in [13].

1.1 Related results

Several papers deal with the approximation of independent sets in bounded degree graphs [16, 8, 17, 13, 15]. Hochbaum [16] gave an algorithm with a $\Delta/2$ performance ratio, which remained unchallenged for over a decade. Her algorithm also applies to node-weighted graphs.

More recently, Berman and Fürer [8] gave a type of a local search algorithm that attains a greatly improved $(\Delta+3)/5+1/\epsilon$ performance ratio. Berman and Fujito [7] eliminated a small additive term in the performance ratio of [8] for odd values of Δ. The disadvantage of their method is extreme time complexity: although improved analysis yields some reduction [15], the time complexity starts at roughly n^{50} and becomes significantly greater if ϵ is desired to be small. Khanna et al. [17] independently obtained an improvement over the ratio of [16], using a simpler local search method.

Our other work on this problem includes a $(\Delta+2)/3$ bound on the *greedy* algorithm [13], and a $(\Delta+3)/4$ bound on an efficient variant of the Berman-Fürer algorithm [15]. A parallel algorithm attaining the former bound is also given in [13].

1.2 Notation

We use fairly standard graph terminology. For the graph in question, usually denoted by G, n denotes the number of vertices, Δ maximum degree, $\overline{d}$ average degree, α independence number (or size of the maximum independent set), and τ independence fraction (or the ratio of the independence number to the number of vertices). For a vertex v, $d(v)$ denotes the degree of v, and $N(v)$ the set of neighbors of v. For a vertex set $S \subseteq V$, $G-S$ denotes the graph induced by vertices in $V-S$. K_ℓ denotes the complete graph (i.e. clique) on ℓ vertices.

For an independent set algorithm Alg, $\mathsf{Alg}(G)$ is the size of the solution obtained on graph G. The performance ratio of the algorithm in question is defined by

$$\rho = \max_G \frac{\alpha(G)}{\mathsf{Alg}(G)}.$$

We consider ρ to be a function of Δ, in which case we can assume the maximum to be taken over graphs with that maximum degree.

2. Subgraph Removal Approach

We present a method for improving the performance ratio of independent set approximation algorithms. Any algorithm whose performance ratio decreases as the independence fraction of the graph decreases can be enhanced using this approach, with greater improvements as the maximum degree gets larger.

The idea is based on the following fact: graphs without dense subgraphs, particularly cliques, contain larger independent sets than graphs do in general, and moreover these larger solutions can be found effectively. We remove all cliques of certain size from the input graph and apply the improved algorithms on the remaining graph. This will be advantageous as long as the input graph contains few disjoint cliques; if it contains many disjoint cliques, the independence number must be low; in either case, the resulting performance ratio will improve.

This schema uses as subroutine two types of algorithms: an approximation algorithm for general graphs, and algorithms that find large independent sets in ℓ-clique-free graphs with possibly different algorithms for different values of ℓ. Better algorithms of either type translate immediately to better performance ratios.

We first illustrate this technique in its simplest form, when removing disjoint 2-cliques (i.e. a matching). We then give the structure of the general schema. Next, via a constructive proof of a graph theorem, we obtain an asymptotically improved $O(\Delta/\log\log\Delta)$ performance ratio. We then introduce two practical algorithms from the literature and use them to obtain improved bounds for graphs of intermediate maximum degree. The proof of the graph theorem is deferred to Section 3.

2.1 Removing edges

The following simple idea was used by [21] and [8] to obtain a fair performance ratio in linear time. We use it here to obtain a quick approximation in terms of average degree that is close to the best ratio known. It can also be implemented in parallel, improving the best such performance ratios known.

We apply two different strategies to find an independent set, and retain the larger result. One is to use the complement of some maximal matching, i.e. the vertices not appearing in the matching. It contains $n-2m$ vertices, where m is the number of matched edges, while the independence number is at most $n-m$. Thus, the size of the solution found is at least $2\alpha - n$, for a performance ratio of $\tau/(2\tau-1)$.

The other is the Greedy algorithm, which iteratively selects a vertex of minimum degree and removes it and its neighbors from the graph. The size of the solution it finds satisfies:

FACT 1. ([13]) $\mathsf{Greedy}(G) \geq \dfrac{1+\tau^2}{\overline{d}+1+\tau}n,$

for an approximation ratio of $(\overline{d}+1+\tau)\tau/(1+\tau^2)$.

Observe that the former ratio is monotone decreasing with τ ($\tau > 1/2$), while the latter ratio is monotone increasing. A close study shows that the value of τ for which the ratios agree is at most $\frac{1}{2} + \frac{5/4}{2\overline{d}+2}$. If we plug that into the higher ratio, that of the maximal-matching complement, this yields a performance ratio of $(2\overline{d}+4.5)/5$. Thus, in linear time, independent of $\overline{d}$, we come within an additive 0.3 of the $(2\overline{d}+3)/5$ performance ratio of [13] that requires $\Omega(\overline{d}n^{3/2})$ time.

Efficient parallel algorithms are also known for finding both a solution satisfying Fact 1 [14] as well as a maximal matching. This combination therefore improves on the previous $(\overline{d}+2)/2$ performance ratio of the former algorithm [14].

The idea of using a maximal matching to upper-bound the optimal solution can be generalized naturally if we think of an edge as a clique on 2 vertices. After removing a maximal collection of disjoint ℓ-cliques, the remaining graph will not be independent, yet will be more amenable to the discovery of large independent subgraphs.

2.2 Generic Clique Removal Schema

We present an algorithm schema, indexed by a cardinal k and a collection of subordinate procedures. One is algorithm General-BDIS-Algorithm for finding independent sets in graphs of bounded-degree that are otherwise unrestricted. The other algorithms are for finding independent sets in ℓ-clique-free graphs, possibly one for each value of ℓ, $2 \leq \ell \leq k$.

```
CliqueRemoval_k(G)
  I ← General-BDIS-Algorithm(G)
  for ℓ ← k downto 2 do
    S ← CliqueCollection(G,ℓ)
    G ← G − S
    I_ℓ ← K_ℓ-free-BDIS-Algorithm(G)
    I ← max (I, I_ℓ)
  od
  return I
end
```

The algorithm CliqueCollection finds in G a maximal collection of disjoint cliques of size ℓ; in other words, S is a set of mutually non-intersecting cliques of size ℓ such that the graph $G - S$ contains no ℓ-cliques. Such a collection can be found in $O(\Delta^{\ell-1}n)$ time by searching exhaustively for an $(\ell-1)$-clique in the neighborhood of each vertex. That is polynomial whenever $\ell = O(\log_\Delta n)$.

When k is 2 and Greedy is used as the general algorithm, the result is the previous edge-removal algorithm. We shall present here two further instances of this schema.

2.3 Asymptotic Improvement

Asymptotically larger independent sets are known to exist in K_ℓ-free graphs.

THEOREM 1. (AJTAI ET AL. [1]) *There exists an absolute constant c_1 such that for any K_ℓ-free graph G,*

$$\alpha(G) \;\geq\; c_1 \frac{\log((\log \overline{d})/\ell)}{\overline{d}} n.$$

By derandomizing the parts of the proof of [1] where probabilistic existence arguments are used, we have obtained an algorithm AEKS that constructs the promised independent set in polynomial time. It, and its analysis, are presented in Section 3.

To obtain improved approximations, it suffices to use the following simplified algorithm to approximate independent sets. An independent set is said to be *maximal* (MIS) if adding any further vertices to the set violates independence. An MIS is easy to find and provides a sufficient, general upper bound of $n/(\Delta+1)$.

```
AEKS-CR(G)
  G' ← G − CliqueCollection(G, c_1 log log Δ)
  return max(AEKS(G'), MIS(G))
end
```

THEOREM 2. *The performance ratio of* AEKS-CR *is* $O(\Delta/\log\log\Delta)$.

PROOF. Let z denote $c_1 \log\log\Delta$, and let n' denote the size of $V(G')$. The independence number collects at most one from each z-clique, yielding

$$\alpha \ \le \ n/z + n' \ \le \ 2\max(n/z, n'),$$

while the size of the solution found by AEKS-CR is at least

$$\mathsf{AEKS\text{-}CR}(G) \ \ge \ \max(\frac{1}{\Delta+1}n, \frac{c_1}{2}\frac{\log\log\Delta}{\Delta}n') \ \ge \ \frac{z/2}{\Delta+1}\max(n/z, n').$$

The ratio between the two satisfies the claim. □

Observe that the combined method runs in polynomial time for Δ as large as $n^{1/\log\log n}$.

2.4 Effective method for moderately large maximum degree

We now turn our attention to practical methods that can benefit from the clique removal schema. This involves an algorithm of Shearer [22] for 3-clique-free graphs, and a simple local search algorithm for other ℓ-clique-free graphs as well as for otherwise unrestricted graphs of bounded-degree. The performance ratio of the resulting algorithm is asymptotically $\Delta/6(1+o(1))$ and improves the best ratios known for moderate to large values of Δ. In order to improve the approximation for small values of Δ, we replace edge-removal by a preprocessing technique of Hochbaum [16].

2.4.1 Preprocessing

Based on a theorem of Nemhauser and Trotter [19] on solutions of the linear programming formulation of the independent set problem, Hochbaum [16] introduced a preprocessing method to improve the performance of approximation algorithms for the problem. The theorem of [19] states that in the time taken to compute a bipartite matching, the vertices of the graph can be partitioned into three sets P, Q, and R with the properties that: i) the vertices in P are not adjacent to any vertex in either P or Q, and ii) the size of the optimal solution is at most $|P| + \frac{1}{2}|Q|$. As a result, we can restrict our attention to the graph induced by Q, and attach the set P to the solution found by the approximation algorithm on that graph. The corollary of this is that in a worst case instance, the independence number is at most half the number of vertices.

2.4.2 2-opt

Khanna et al. [17] studied a simple local search algorithm that we have named 2-opt. Starting with an initial maximal independent set (MIS), it tries all possible ways of increasing the independent set solution by adding two vertices while removing only one. Given an independent set I, and vertices $v_1, v_2 \in V - I$, $u \in I$, the triple $\langle v_1, v_2, u\rangle$ is said to be a *2-improvement* of I if the symmetric difference $I \oplus \{v_1, v_2, u\} = (I - \{u\}) \cup \{v_1, v_2\}$ is independent. Since I can be assumed to be a maximal independent set, it suffices to look at pairs adjacent to a common vertex in I.

```
2-opt(G)
  I ← MIS(G)
  while (∃ 2-improvement ⟨v1, v2, u⟩ of I)
    I ← I ∪ {v1, v2} − {u}
  return I
end
```

The algorithm clearly runs in polynomial time, and can be implemented in linear time for graphs of constant degree. We can use it to obtain good bounds on graphs with high independence fraction.

LEMMA 1. (KHANNA ET AL. [17]) 2-opt(G) $\geq \dfrac{1+\tau}{\Delta+2}n$.

PROOF. Consider any independent set I for which no 2-improvement exists. Let B be some maximum independent set of G, and C be the intersection of B and I. Since I is maximal, each vertex in $V - I$ must be adjacent to at least one vertex in I. If two vertices in $B - C$ are adjacent to the same vertex in $I - C$ and neither is adjacent to any other vertex in $I - C$, then the three vertices form a 2-improvement of I. Hence, at least $|B-C|-|I-C| = |B|-|I|$ vertices are adjacent to two vertices in I. Considering that the degree of vertices in I is at most Δ, we have that $|I|\Delta \geq |V-I|+|B|-|I| = n+\alpha-2|I|$, which yields the the lemma. □

A $\Delta/2.44(1+o(1))$ performance ratio for the combination of this algorithm with Greedy was proved in [17]. An improved bound of $(\Delta+2)/3$ can be obtained via the $\tau \leq 1/2$ promise of the preprocessing technique of [16].

We can obtain improved bounds for ℓ-clique-free graphs.

LEMMA 2. *On a ℓ-clique-free graph G,* $\text{2-opt}(G) \geq \dfrac{2}{\Delta+\ell}n$.

PROOF. If some $\ell-1$ vertices are adjacent to any given vertex $u \in I$ and no other vertex in I, then either they form a ℓ-clique with u or they include a non-adjacent pair, which with u forms a 2-improvement of I. It therefore follows that at most $(\ell-2)|I|$ vertices are adjacent to exactly one vertex in I. Also, since I is maximal, all vertices outside of I must be adjacent to at least one vertex in I. Summing up the number of edges incident on I, we have that $|I|\Delta \geq |V-I| + (|V-I| - (\ell-2)|I|) = 2n - \ell|I|$, which yields the lemma. $\square$

The above lemma appears also in [10] with a similar proof as a bound on the independence number in terms of maximum degree, clique number, and vertex number.

2.4.3 Shearer

A classical theorem of graph theory due to Turán states that the independence number of a graph is at least $n/(\overline{d}+1)$. In fact, this bound is attained by the *Greedy* algorithm mentioned earlier (see [16, 14]). Ajtai, Komlós, and Szemerédi [2] obtained the first asymptotically improved bound for graphs no cliques on three vertices. (Note that Theorem 1 is a generalization to larger forbidden cliques, albeit with weaker conclusion.) It was improved by Shearer to the following theorem.

THEOREM 3. (SHEARER [22]) *Let $f_s(d) = (d \ln d - d + 1)/(d-1)^2$, $f_s(0) = 1$, $f_s(1) = \frac{1}{2}$. For a triangle-free graph G, $\alpha(G) \geq f_s(\overline{d})n$.*

Moreover, he gave the following algorithm that attains the claimed bound. Let f_s' denote the derivative of f_s.

Shearer(G)
  $I \leftarrow \emptyset$
  $H \leftarrow G$
  while $H \neq \emptyset$ do
    Pick a vertex v of degree d_v such that
      $(d_v+1)f_s(\overline{d}(H)) \;\leq\; 1+(\overline{d}d_v+\overline{d}-2\sum_{w\in N(v)} d(w))f_s'(\overline{d}(H))$
    $I \leftarrow I \cup \{v\}$
    $H \leftarrow H - (N(v) \cup \{v\})$
  od
  return I
end

In fact, the claim is also satisfied in linear time, independent of Δ, by a simple randomized greedy algorithm, if we modify the above algorithm to select a random vertex in each step [22]. Shearer later gave a slightly improved function [23].

2.4.4 Analysis

We now analyze the algorithm that results from instantiating the clique removal schema with 2-opt and Shearer, and additionally applying preprocessing on the triangle-free graphs. We obtain the following explicit, if less than compact, bound on the performance ratio. Let $\mathcal{H}_k$ denote the k-th Harmonic number $\sum_{i=1}^{k} 1/i$.

THEOREM 4. CliqueRemoval$_k$*, using* 2-opt*,* Shearer*, and preprocessing, attains a performance ratio of at most*

$$\left[\frac{\Delta}{2} + 2 + \frac{k}{2}\left(\mathcal{H}_{k-1} + \frac{1}{3f_s(\Delta)} - \frac{3}{2} + \frac{\Delta}{3}\right)\right] / (k+1)$$

for graphs of maximum degree $\Delta \geq 5$.

PROOF. Let n_ℓ denote the number of vertices in the ℓ-clique-free graph, $3 \leq \ell \leq k$. Let n_2 denote the size of the set P found by preprocessing. Thus, $n \geq n_k \geq \ldots \geq n_3 \geq n_2 \geq 0$.

The size of the optimal solution is τn, which can be bounded by

$$\tau n \;\leq\; n_2 + \tfrac{1}{2}(n_3 - n_2) + \cdots + \tfrac{1}{k}(n - n_k) \;=\; \sum_{\ell=2}^{k} \frac{1}{\ell(\ell-1)} n_\ell + \frac{1}{k} n. \quad (1)$$

By Lemmas 1 and 2, Theorem 3, and the property of preprocessing, our algorithm is guaranteed to output a solution of size at least

$$\max\left[\frac{1+\tau}{\Delta+2} n,\ \max_{4 \leq \ell \leq k} \frac{2}{\Delta+\ell} n_\ell,\ n_2 + f_s(\Delta) n_3\right],$$

and its performance ratio is therefore bounded by

$$\rho \;\leq\; \min\left[\frac{\tau n}{\frac{1+\tau}{\Delta+2} n},\ \min_{4 \leq \ell \leq k} \frac{\tau n}{\frac{2}{\Delta+\ell} n_\ell},\ \frac{\tau n}{n_2 + f_s(\Delta) n_3}\right].$$

From this we derive, respectively, that

$$\tau \;\geq\; \frac{\rho}{\Delta + 2 - \rho}, \quad (2)$$

$$n_\ell \;\leq\; \frac{\tau}{\rho} \frac{\Delta+\ell}{2} n, \quad \ell = 4, 5, \ldots, k \quad (3)$$

$$n_2 + n_3 \;\leq\; \frac{\tau}{\rho} \frac{1}{f_s(\Delta)} n. \quad (4)$$

Combining (1), (4) and (3), we find that

$$\tau \;\leq\; \frac{\tau}{\rho} s_{\Delta,k} + \frac{1}{k}, \tag{5}$$

where

$$s_{\Delta,k} \;=\; \frac{1}{6f_s(\Delta)} + \sum_{i=4}^{k} \frac{\Delta + i}{2i(i-1)} \;=\; \frac{1}{2}\left[\frac{1}{3f_s(\Delta)} + (\mathcal{H}_{k-1} - \frac{3}{2}) + \Delta(\frac{1}{3} - \frac{1}{k})\right].$$

Multiplying (5) by ρ/τ, and bounding τ by (2), we obtain

$$\rho \;\leq\; s_{\Delta,k} + \frac{\rho}{\tau k} \;\leq\; s_{\Delta,k} + \frac{\Delta + 2 - \rho}{k}$$

which, when rearranged, yields the desired

$$\rho \;\leq\; \frac{\Delta + 2 + k s_{\Delta,k}}{k+1}\;. \;\square$$

We can from this compute the ratio for particular values of Δ. It is also easy to see that if Δ and k are large, then the $\Delta/3$ term in $s_{\Delta,k}$ will dominate for a $\Delta/6$ asymptotic ratio.

2.4.5 Comparison

We compare the ratios guaranteed by CliqueRemoval to those of recently analyzed algorithms in Table 1. Its strengths are in the higher values of Δ, surpassing the best ratio of a low-polynomial time algorithm for $\Delta \geq 33$, and the best ratio for any polynomial time algorithm [8] for $\Delta \geq 613$.

Δ	CliqRem	Berman-Fürer [8]	2Δ-opt[15]	Greedy [13]
10	3.54	2.60	3.25	4.00
33	8.92	7.25	9.00	11.66
100	23.01	20.60	25.75	34.00
1024	201.57	205.40	256.75	342.00
8192	1535.20	1639.00	2048.75	2731.33

TABLE I: Performance ratios of recent independent set algorithms.

3. Constructive Proof of Combinatorial Theorem

In this section, we give an algorithm that attains Theorem 1, restated here for convenience.

THEOREM. (AJTAI ET AL. [1]) *There exists an absolute constant c_1 such that any graph on n vertices not containing an induced K_ℓ has an independent set of size at least $c_1(n/\overline{d})\log A$, where $A = (\log \overline{d})/\ell$.*

Notation

We use the following additional graph notation. For a graph H, let $n(H)$, $e(H)$, and $h(H)$ denote the number of vertices, edges, and triangles, respectively, and let $\overline{d}(H)$ denote the average degree of H. Let $deg_3(v)$ denote the triangle degree of a vertex v, i.e. the number of mutually adjacent vertex triples that include v.

3.1 Few Triangles

As an intermediate step in the proof we need to consider graphs with few triangles.

In [2], Ajtai, Komlós, and Szemerédi extend their theorem for triangle-free graphs to the case where the graph has few triangles. This result is used later in [1] to obtain the result for K_ℓ-free graphs. In this section, we describe how one can efficiently obtain an independent set of size promised by the following theorem.

THEOREM 5. (AJTAI ET AL. [2]) *If the number h of triangles in a graph G is less than $\epsilon n \overline{d}^2$, where $\epsilon > 1/\ln \overline{d}$, then (for $c_2 = 0.001$)*

$$\alpha(G) > c_2(n/\overline{d})\ln(1/\epsilon).$$

In [2], the following procedure is used to establish Theorem 5. (Shearer [22] also proves a version of Theorem 5, but we were unable to extract an efficient algorithm from his proof.)

Set $p = 1/(\sqrt{12\epsilon} \cdot \overline{d})$. Assume that $n(G)p \geq 20$.

AKS(G)
  Obtain an induced subgraph G' with $n' = n(G') > np/2$,
    $h' = h(G') < 3hp^3$, and $e' = e(G') < 3e(G)p^2$.
  $G'' \leftarrow G'$
  for each triangle in G'' do
    delete one vertex of that triangle
  return Shearer(G'')
end

Note that G'' has at least $n' - h' > np/4$ vertices and at most e' edges; thus $\overline{d}(G'') \leq 12\overline{d}(G)p$. Observe that Shearer($G''$) yields an independent set of size at least

$$n(G'')\frac{\ln 12\overline{d}(G)p}{24\overline{d}(G)p} \geq c_2(n/\overline{d}(G))\ln(1/\epsilon),$$

for c_2 chosen sufficiently small.

In the original proof, the subgraph G' in was obtained using a probabilistic existential argument. We now show how it can be implemented efficiently. First, we describe the probabilistic method.

Let G' be obtained by putting each $v \in V(G)$ in $V(G')$ with probability p, these events being 3-wise independent. Then

$$E[h(G')] = h(G)p^3, \quad E[e(G')] = e(G)p^2$$
$$E[n(G')] = n(G)p, \quad \mathrm{var}[n(G')] = n(G)p(1-p).$$

Then, we have using Chebyshev's inequality that

$$\Pr[n(G') < n(G)p/2] < \frac{4(1-p)}{np} \;\leq\; \frac{1}{5},$$

and, using Markov's inequality, that

$$\Pr[e(G') \geq 3e(G)p^2] \;\leq\; \frac{1}{3},$$

and

$$\Pr[h(G') \geq 3h(G)p^3] \;\leq\; \frac{1}{3}.$$

Since these probabilities add up to less than 1, the required subgraph does exist. To obtain this subgraph, we shall construct a set of sequences of choices so that the events $v \in V(G')$ are 3-wise independent. Thus one of the sequences in the set will correspond to the required graph G'. We will then be able to obtain the graph explicitly by trying all sequences. This is reasonable because the size of the set needed for this will be small.

Identify the vertex set of G with $\{0, 1, \ldots, n-1\}$ and let q be a prime such that $n \leq q \leq 2n$. Let D be a subset of $\{0, 1, \ldots, q-1\}$ of size $\lceil pq \rceil$. For $a, b, c \in \{0, 1, \ldots, q-1\}$ define the subgraph $G'(a, b, c)$ by letting

$$v \in V(G'(a,b,c)) \;\text{ iff }\; av^2 + bv + c \pmod{q} \;\in\; D.$$

For a, b, c chosen independently and uniformly from $\{0, 1, \ldots, q-1\}$, it can be shown that

$$p' \doteq \Pr[v \in V(G')] = \frac{|D|}{q} = \frac{\lceil pq \rceil}{q},$$

and that the events $\{v \in V(G'(a,b,c))\}_{v \in V(G)}$ are 3-wise independent [18, p. 200]. Note that $p \leq p' \leq p(1 + 1/q)$. Arguing as above, we can show that

$$\Pr[n(G'(a,b,c)) < n(G)p/2] \;<\; \frac{4(1-p')}{np'} \;\leq\; \frac{1}{5},$$

$$\Pr[e(G'(a,b,c)) \geq 3e(G)p^2] \;\leq\; \frac{p'^2}{3p^2} \;\leq\; \frac{1}{3}\left(1 + \frac{1}{pq}\right)^2 \;\leq\; 0.3675,$$

and

$$\Pr[h(G'(a,b,c)) \geq 3h(G)p^3] \;\leq\; \frac{p'^3}{3p^3} \;\leq\; \frac{1}{3}\left(1 + \frac{1}{pq}\right)^3 \;\leq\; 0.385875.$$

Note that these probabilities add up to less than 1. To construct the graph G' we check for all possible triples $(a, b, c) \in \{0, 1, \ldots, q-1\}^3$ whether $G'(a, b, c)$ has the required properties. The discussion above shows that there exists a choice of (a, b, c) such that the graph $G'(a, b, c)$ has the desired properties.

3.2 Graphs with no K_ℓ

We are now ready to present the method to produce an independent set in K_ℓ-free graphs of size promised by Theorem 1.

LEMMA 3. *Let $\ell \geq 2$, $0 < \delta < 1/2$ be arbitrary. If a graph H contains no K_ℓ, then it contains a (spanned) subgraph H' with*

$$n(H') \geq (2\delta)^{\ell-2} n(H), \quad e(H') < \delta(n(H'))^2.$$

It is shown in [1] that the following procedure constructs the desired subgraph H'.

SparseSubgraph(H, ℓ, δ)
  if $\ell = 2$ return H
  if $e(H') < \delta(n(H'))^2$ then return H'
  choose $v \in V(H)$ with $d(v) > 2\delta n(H)$
  return SparseSubgraph$(N(v), \ell - 1, \delta)$
end

LEMMA 4. *If H contains no K_ℓ, then it can be partitioned into $H = H_0 \cup H_1 \cup H_2 \cup \ldots$ in such a way that*

$$n(H_i) = \left\lfloor \delta^{\ell-1} n(H) \right\rfloor, \quad e(H_i) < \delta(n(H_i))^2, \quad i = 1, 2, \ldots,$$

and for the leftover H_0, $n(H_0) < \delta n(H)$.

The proof of this lemma given in [1] uses the following procedure.

Partition(H, δ)
  $i \leftarrow 1$
  $H^* \leftarrow H$
  while $n(H^*) \geq \delta n(H)$ do
    $H' \leftarrow$ SparseSubgraph$(H, \ell, \delta/2)$
    $H_i \leftarrow$ subgraph of H' such that
      $n(H_i) = \lfloor \delta^{\ell-1} n(H) \rfloor$ and $e(H_i) < \delta(n(H_i))^2$
    $H^* \leftarrow H^* - H_i$
    $i \leftarrow i + 1$
  od
  $H_0 \leftarrow H^*$
  return $(H_0, H_1, \ldots)$
end

The description given above is incomplete because we have not described how the subgraph H_i can be obtained. In the original proof this was obtained using a straightforward application of the probabilistic method. This approach can be derandomized using the *method of conditional expectation* [4]. We defer the details until later, for we shall soon use this method for another somewhat complicated instance.

And now, finally, we have the procedure for constructing an independent set of the size promised by Theorem 1. The proof in [1] uses the following procedure.

Let $A = (\log \overline{d})/\ell$ as in the statement of the theorem. Define $\epsilon = A^{-3c_1/c_2}$, and $\delta = \epsilon/10$.

```
InitialPartition(G)
  m ← ⌊n/Δ⌋
  for i ← 1 to m do
    v ← vertex of largest triangle degree deg3(v) in G
    V_i ← {v} ∪ N(v) ∪ {some Δ − (d(v) + 1) additional vertices}
    G ← G − V_i
  od
  V_{m+1} ← V(G)
  return (V_1, V_2, ..., V_m, V_{m+1})
end
```

Let d_0 be the smallest degree for which Theorem 1 improves on Turán's bound, i.e. roughly $\exp(\exp(1/c_1))$.

```
AEKS(G)
  if d̄ ≤ d_0 then return Turán(G)
  if Δ > d̄ + 10d̄/(log d̄) then
      let v be a vertex of degree Δ
      return AEKS(G − {v})

  (V_1, V_2, ..., V_m, V_{m+1}) ← InitialPartition(G)
  G' ← subgraph induced by V_{m/2+1} ∪ V_{m/2+2} ∪ ... ∪ V_m
  if (deg3(G) < εΔ^2) then return AKS(G')

  for i ← 1 to m do
      G_i ← subgraph induced by V_i
      (V_{i0}, V_{i1}, ..., V_{it_i}) ← Partition(G_i, δ)
  od
  e* ← the number of edges with ends in different V_i
  j_i ← a choice in {1, 2, ..., t_i} s.t. the number of edges between
         the subclasses {V_{ij_i}}_{i=1}^m is at most e* δ^{2ℓ−2}/(1 − δ)^2.
  G'' ← the subgraph induced by ⋃_i V_{ij_i}
  return AEKS(G'')
end
```

3.3 Obtaining the choice function

It remains to describe how the choice function j_i used above can be obtained. We wish to present a derandomized version of the averaging argument used in the original proof. To make the presentation simple we first consider the following problem, whose solution we shall use later to obtain the choice function.

Suppose we have a graph G with $V(G)$ partitioned as $V_1 \cup V_2 \cup \ldots \cup V_m$, such that no edge has both its ends in any V_i. Further assume that each edge e has an integer weight $\mathrm{wt}(e)$ and each vertex has weight $p(v) \in [0,1]$, such that, for each i, $\sum_{v \in V_i} p(v) = 1$. For $i = 1, 2, \ldots, m$, let X_i be the random variable defined by

$$\Pr[X_i = v] = p(v), \quad \text{for} \quad v \in V_i,$$

the X_i's being mutually independent. Let G' be the (random) subgraph of G induced by $\{X_1, X_2, \ldots, X_m\}$. Let $W(G')$ be the sum of the weight of the edges of G'. Then, we have

$$\begin{aligned} E[W] &= \sum_{\{v,w\} \in E(G)} \mathrm{wt}(\{v,w\}) \Pr[\{v,w\} \in E(G')] \\ &= \sum_{\{v,w\} \in E(G)} \mathrm{wt}(\{v,w\}) p(v) p(w). \end{aligned}$$

Clearly, then, there exists a subgraph G^* of G consisting of exactly one vertex from each V_i, such that $W(G^*) \le E[W]$. We now show that such a graph G^* can be constructed, provided $\mathrm{wt}(\cdot)$ and $p(\cdot)$ are given.

The following definition will play a central role in our algorithm. Let $1 \le r \le m$ and $v_i \in V_i$, for $i = 1, 2, \ldots, r$. Then

$$\begin{aligned} E[W \mid (v_1, v_2, \ldots, v_r)] &\doteq E[W \mid \bigwedge_{i=1}^{r} X_i = v_i] \\ &= \sum_{e \in E(G)} \mathrm{wt}(e) \Pr[e \in E(G') \mid \bigwedge_{i=1}^{r} X_i = v_i]. \end{aligned}$$

For $e = \{v, w\} \in E(G)$, observe that

$$\begin{aligned} &\Pr[e \in E(G') \mid \bigwedge_{i=1}^{r} X_i = v_i] \\ &= \begin{cases} 1 & \text{if } v, w \in \{v_1, v_2, \ldots, v_r\} \\ p(w) & \text{if } v \in \{v_1, v_2, \ldots, v_r\} \text{ and } w \in \bigcup_{i=r+1}^{m} V_i \\ p(v)p(w) & \text{if } v, w \in \bigcup_{i=r+1}^{m} V_i \\ 0 & \text{otherwise.} \end{cases} \end{aligned}$$

It follows, then, that if we are given $v_1, v_2, \ldots, v_r$ then we can easily compute $E[W \mid (v_1, v_2, \ldots, v_r)]$. Further, for $r < m$,

$$E[W \mid (v_1, v_2, \ldots, v_r)] = \sum_{v \in V_{r+1}} p(v) E[W \mid (v_1, v_2, \ldots, v_r, v)]. \qquad (6)$$

We now present the algorithm to find the graph G^*.

```
ChoiceGraph(G)
  for i ← 1 to m do
    v_i ← v ∈ V_i minimizing E[W | (v_1, v_2, ..., v_{i-1}, v)]
  return the subgraph G* induced by (v_1, v_2, ..., v_m)
end
```

It follows by an easy induction on i, using (6), that $E[W \mid (v_1, v_2, \ldots, v_i)] \le E[W]$, and, in particular, $E[G'] \le E[W]$.

3.3.1 The choice function

To obtain the choice function used in the procedure AEKS we consider the graph $\hat{G}$ with the vertex sets V_i defined by

$$V_i = \{(i, j) : j = 1, 2, \ldots, t_i\},$$

the weight of edge $\{(i_1, j_1), (i_2, j_2)\}$ set to be the number of edges between vertex sets $V_{i_1 j_1}$ and $V_{i_2 j_2}$, and

$$p((i, j)) = \frac{1}{t_i}.$$

Consider the invocation of Partition(G_i, δ) in the procedure AEKS. We have (Lemma 4) $|V_{i0}| \le |V_i|$ and, for $j \ge 1$, $|V_{ij} \le \delta^{l-1}|V_i|$. Thus, $t_i \ge (1-\delta)/\delta^{\ell-1}$, and hence $p(v) \le \delta^{\ell-1}/(1-\delta)$ for all $v \in V(\hat{G})$. Using the procedure above we may obtain vertices $\{(i, j_i)\}_{i=1}^m$ such that the sum of the weights of edges in the graph induced by them is at most

$$E[W] \le \frac{\delta^{2\ell-2}}{(1-\delta)^2}\, e^*.$$

This gives us the required choice function.

3.4 Obtaining the Sparse Subgraph

Now we return to the application of the probabilistic method in the procedure Partition. This time we have a graph G on n vertices, and we consider the random subgraph H induced by a random subset of $V(G)$ of size k. Then,

$$E[e(H)] = \sum_{e \in E(G)} \frac{k(k-1)}{n(n-1)} = e(G)\frac{k(k-1)}{n(n-1)} < (k^2/n^2) e(G).$$

Clearly, there exists a subgraph $\hat{H}$ with k vertices and at most $E[e(H)]$ edges. We show now how such a subgraph can be obtained efficiently.

Let $S \subseteq V(G)$ and $|S| = s \leq k$. Then

$$E[e(H) \mid S \subseteq V(H)] = \sum_{e \in E(G)} \Pr[e \in E(H) \mid S \subseteq V(H)].$$

Note that

$$\Pr[e \in E(H) \mid S \subseteq V(H)] = \begin{cases} 1 & \text{if } |e \cap S| = 2 \\ \frac{k-s}{n-s} & \text{if } |e \cap S| = 1 \\ \frac{(k-s)(k-s-1)}{(n-s)(n-s-1)} & \text{if } |e \cap S| = 0. \end{cases}$$

Thus, for any set S we can easily compute $E[e(H) \mid S \subseteq V(H)]$. Also,

$$E[e(H) \mid S \subset V(H)] = \sum_{v \in V(G)-S} \frac{1}{n-s} E[e(H) \mid S \cup \{v\} \subseteq V(H)] \qquad (7)$$

We then have the following algorithm for finding the subgraph $\hat{H}$.

FindSubgraph(H)
  $S_0 \leftarrow \emptyset$
  for $i \leftarrow$ 1 to k do
    $v_i \leftarrow v \in V - S_{i-1}$ minimizing $E[e(H) \mid S_{i-1} \cup \{v\} \subseteq V(H)]$
    $S_i \leftarrow S_{i-1} \cup \{v_i\}$
  od
  return the subgraph $\hat{H}$ induced by S_k
end

It follows by a routine induction, using (7), that $E[e(H) \mid S_i \subseteq V(H)] \leq E[e(H)]$, and, in particular, that $\hat{H}$ has the required properties.

Let us apply this method to the step in the procedure Partition. For our application there, we take G to be H' and k to be $\lfloor \delta^{\ell-1} n(H) \rfloor$. This completes our treatment of the proof of Theorem 1.

4. Discussion

The clique removal schema links further improvements in approximation to open questions in graph theory. If an $\Omega(n(\log \Delta)/\Delta)$-independent set can be found in $(\log \Delta)$-clique-free graphs, as has been conjectured, a performance ratio of $O(\Delta/\log \Delta)$ would ensue. On the other hand, this appears to be the limit of a direct application of this technique. A related question is whether ℓ-clique-free graphs can be colored with $o(\Delta)$ colors; an affirmative resolution would assist in approximating independent sets in weighted graphs.

For small values of Δ, various improvements are plausible with improved component algorithms. A stronger bound for Shearer when the independence number is large – similar to what we obtained for Greedy [13] – would be particularly useful.

Acknowledgements

We would like to thank an anonymous referee for detailed comments that helped improve the presentation of the paper.

References

[1] M. Ajtai, P. Erdős, J. Komlós, and E. Szemerédi. On Turán's theorem for sparse graphs. *Combinatorica*, 1(4):313–317, 1981.
[2] Miklós Ajtai, János Komlós, and Endre Szemerédi. A note on Ramsey numbers. *J. Combin. Theory Ser. A*, 29:354–360, 1980.
[3] Noga Alon, Uriel Feige, Avi Wigderson, and David Zuckerman. Derandomized graph products. Manuscript, September 1994.
[4] Noga Alon and Joel H. Spencer. *The Probabilistic Method*. Wiley, 1992.
[5] Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. Proof verification and hardness of approximation problems. In *Proc. 33rd IEEE Symp. on Found. of Comp. Sci.*, pages 14–23, October 1992.
[6] Mihir Bellare and Madhu Sudan. Improved non-approximability results. In *Proc. 26th Ann. ACM Symp. on Theory of Computing*, pages 184–193, May 1994.
[7] Piotr Berman and Toshihiro Fujito. Approximating independent sets in degree 3 graphs. Manuscript, 1994.
[8] Piotr Berman and Martin Fürer. Approximating maximum independent set in bounded degree graphs. In *Proc. Fifth ACM-SIAM Symp. on Discrete Algorithms*, pages 365–371, January 1994.
[9] Ravi B. Boppana and Magnús M. Halldórsson. Approximating maximum independent sets by excluding subgraphs. *BIT*, 32(2):180–196, June 1992.
[10] Robert C. Brigham and Ronald D. Dutton. A compilation of relations between graph invariants. *Networks*, 15:73–107, 1985.
[11] Paul Erdős. Some remarks on chromatic graphs. *Colloq. Math.*, 16:253–256, 1967.
[12] Uriel Feige. Private communication, 1994.
[13] Magnús M. Halldórsson and Jaikumar Radhakrishnan. Greed is good: Approximating independent sets in sparse and bounded-degree graphs. In *Proc. 26th Ann. ACM Symp. on Theory of Computing*, pages 439–448, May 1994.
[14] Magnús M. Halldórsson and Jaikumar Radhakrishnan. Greed is good: Approximating independent sets in sparse and bounded-degree graphs. Submitted, 1994.
[15] Magnús M. Halldórsson and Jaikumar Radhakrishnan. Improved approximations of independent sets in bounded-degree graphs. In *Proc. Fourth Scand. Workshop on Algorithm Theory*, pages 195–206. Springer LNCS #824, 1994.
[16] Dorit S. Hochbaum. Efficient bounds for the stable set, vertex cover, and set packing problems. *Disc. Applied Math.*, 6:243–254, 1983.
[17] Sanjeev Khanna, Rajeev Motwani, Madhu Sudan, and Umesh Vazirani. On syntactic versus computational views of approximability. In *Proc. 35th IEEE Symp. on Found. of Comp. Sci.*, 1994.
[18] Dexter Kozen. *The Design and Analysis of Algorithms*. Springer-Verlag, 1992.
[19] G. L. Nemhauser and L. E. Trotter. Vertex packings: Structural properties and algorithms. *Math. Programming*, 8:232–248, 1975.
[20] Christos H. Papadimitriou and Mihalis Yannakakis. Optimization, approximation, and complexity classes. *Journal of Computer and System Sciences*, 43:425–440, 1991.
[21] V. Th. Paschos. A $(\Delta/2)$-approximation algorithm for the maximum independent set problem. *Information Processing Letters*, 44:11–13, November 1992.
[22] James B. Shearer. A note on the independence number of triangle-free graphs. *Discrete Math.*, 46:83–87, 1983.
[23] James B. Shearer. A note on the independence number of triangle-free graphs, II. *J. Combin. Theory Ser. B*, 53:300–307, 1991.