Greed is Good: Approximating Independent Sets in Sparse and Bounded-degree Graphs

Magnús M. Halldórsson
School of Information Science
Japan Adv. Institute of Science and Tech.
Ishikawa 923-12, JAPAN
magnus@jaist.ac.jp

Jaikumar Radhakrishnan
Theoretical Computer Science Group
Tata Institute of Fundamental Research
Bombay, India
jaikumar@tifrvax.bitnet

Abstract

The *minimum-degree Greedy* algorithm, or Greedy for short, is one of the simplest, most efficient, and most thoroughly studied methods for finding independent sets in graphs. We show that it surprisingly achieves a performance ratio of $(\Delta+2)/3$ for approximating independent sets in graphs with degree bounded by Δ. The analysis directs us towards a simple parallel and distributed algorithm with identical performance, which on constant-degree graphs runs in $O(\log^* n)$ time using linear number of processors. We also analyze the Greedy algorithm when run in combination with a fractional relaxation technique of Nemhauser and Trotter, and obtain an improved $(2\overline{d}+3)/5$ performance ratio on graphs with average degree $\overline{d}$.

Finally, we introduce a generally applicable technique for improving the approximation ratios of independent set algorithms, and illustrate it by improving the performance ratio of Greedy for large Δ.

1 Introduction

An *independent set* in a graph is a collection of vertices that are mutually non-adjacent. The problem of finding an independent set of maximum cardinality is one of the fundamental combinatorial problems. Unfortunately, it is known to be $\mathcal{NP}$-complete, even for bounded-degree graphs, and therefore no efficient algorithms are in sight.

Given the hardness of exact computation, we are interested in approximation algorithms for the independent set problem in bounded-degree graphs. In particular, we seek an algorithm with a good *performance ratio*, which is a bound on the maximum ratio between the optimal solution size (i.e. the independence number) and the size of the solution found by the heuristic. The study of such algorithm has become increasingly more prevalent.

One of the most ubiquitous heuristic methods for this problem is the *greedy algorithm*. It iteratively selects a vertex of minimum degree and deletes that vertex and all of its neighbors from the graph, until the graph becomes empty. As a delightfully simple and efficient algorithm, the Greedy method deserves a particularly detailed analysis. It is already known to possess several important properties: attaining the Turán bound, and its generalization in terms of degree sequences [7]; almost always obtaining a solution at least half the size of an optimal solution in a general random graph [2]; yielding a non-trivial graph coloring approximation [12] when applied iteratively; and finding optimal independent sets in forests, series-parallel graphs, and cographs. While Greedy has been frequently studied before, the true extent of its performance ratio has apparently not been determined previously. The best ratio previously claimed was $\Delta-1$ on graphs with maximum degree Δ [20] and $\overline{d}+1$ on graphs of average degree $\overline{d}$.

Our main result is that Greedy is surprisingly much better than previously expected. We obtain a tight performance ratio of $(\Delta+2)/3$ in terms of maximum degree, and an asymptotically optimal bound of $(\overline{d}+2)/2$ in terms of average degree. It comes as a considerable surprise that this simple, linear time method performs as well as it does. In the process, we give a natural extension of Turán's bound that incorporates the actual independence number of the graph, and give a general, tight expression of the size of the solution found as a function of the independence number and the number of vertices. Section 2 contains the detailed analysis of Greedy.

We further analyze Greedy in combination with a fractional relaxation technique of Nemhauser and Trotter [15, 11], in subsections 2.5 and 2.6. We use it to

STOC 94- 5/94 Montreal, Quebec, Canada
© 1994 ACM 0-89791-663-8/94/0005..$3.50

improve the best performance ratio known in terms of average degree to $(2\overline{d}+3)/5$, but show it to be of limited use in terms of maximum degree.

To further improve the performance of Greedy, we introduce in section 3 a new technique that involves removing small dense subgraphs. This technique is a general schema that can potentially be applied to any approximation algorithm for this and related problems. By removing all cliques of fixed size from the graph, we can either find a larger solution or obtain a better upper bound on the size of the optimal solution. Using triangles, the performance ratio of Greedy can be improved to $\Delta/3.5$ plus some constant independent of Δ. Removing larger cliques gives gradual improvement to asymptotically $\Delta/3.76$. An improved performance ratio for the vertex cover problem in bounded degree graphs also follows.

Finally, in section 4 we show that the performance ratios proved for Greedy in bounded-degree graphs can also be obtained by a simple parallel algorithm. Our analysis of Greedy suggests that globally minimum degree is not required, in fact, any vertex satisfying the locally evaluated property *"degree of vertex is at most the average of its neighbors' degrees"* can be selected. This is a simple *local rule* that can be implemented efficiently both in parallel and distributed.

1.1 Related results

Until very recently, the best ratio claimed for any approximation algorithm for independent sets in bounded degree graphs was $\Delta/2$ [11]. Suddenly, several developments took place.

Berman and Fürer [3] obtained significantly improved performance ratios of $(\Delta+3)/5+\epsilon$ and $(\Delta+3.25)/5+\epsilon$ for even and odd degrees respectively. Their method is a type of a local improvement method. It's drawback is the astoundingly huge complexity of $n^{32\Delta^{4/\epsilon}}$, which for typical solution quality means on the order of $n^{2^{100}}$. While some improvements in the complexity are possible [10], even a n^{50} complexity appears out of reach. Khanna et al. [13] considered a very simple (and fast) local improvement method. Using their analysis, it is easy to show that the performance of that algorithm when complemented with Nemhauser-Trotter is $(\Delta+2)/3$, same as Greedy.

We have since been able to obtain several improved ratios [10]. With the subgraph removal techniques of this paper, we have obtained an asymptotic improvements for large values of Δ to a performance ratio of $O(\Delta/\log\log\Delta)$, and a $\Delta/6(1+o(1))$ ratio for intermediate values of Δ. Also, using the local improvement method of [3], but using only $fn(\Delta)\cdot n$ time, we can obtain a $(\Delta+3)/4$ ratio.

In spite of these related results, we believe the results on Greedy reported here hold their own, given the algorithm's simplicity, superior complexity, and general applicability.

In sparse graphs, the only result we are aware of is a $(\overline{d}+1)/2$ performance ratio of Greedy with Nemhauser and Trotter's method [11]. No previous parallel approximation algorithms were known to us, except the Δ and $\delta+1$ ratios that can be obtained from parallel implementations of Brooks' (see [16] for references) and Turán's theorems [9], respectively.

1.2 Notation

We use fairly standard graph notation and terminology. For the graph in question, usually denoted by G, n denotes the number of vertices, Δ the maximum degree, $\overline{d}$ the average degree, α the independence number (the size of the largest independent set), and τ the independence fraction (i.e. α/n). For a vertex v, $d(v)$ denotes the degree of v, and $N(v)$ the set of neighbors of v.

For an independent set algorithm A, the performance ratio of A is defined by

$$\rho_A = \max_G \frac{\alpha(G)}{A(G)}$$

where $A(G)$ is the size of the solution obtained by algorithm A on graph G. We particularly consider two algorithms: Greedy, for short Gr, and the combination of Greedy and Nemhauser-Trotter, denoted by $Gr+NT$. Gr is also a shorthand for $Greedy(G)$.

2 Analysis of Greedy

The Minimum-Degree Greedy algorithm, or Greedy for short, operates as follows. It executes a sequence of *reductions*, each of which corresponds to an iteration, where a vertex is selected and added to the solution, and then it and its neighborhood are removed from the graph. It stops when the graph has been exhausted and outputs the set of selected vertices.

```
Greedy(G)
  I ← ∅
  while G ≠ ∅ do
    Choose v such that d(v) = min_{w∈V(G)} d(w)
    I ← I ∪ {v}
    G ← G − {v} ∪ N(v)
  od
  Output I
end
```

The algorithm can be implemented in time linear in the number of edges and vertices, independent of degree. This involves maintaining a multiset of small non-negative integers (the degrees of the vertices), along

with the associated vertex number, under the operations of unit decrease, deletion, and finding the minimum. This could be implemented by an array of linked lists, one for each degree value, along with an appropriate array structure to provide for direct referencing.

We use the following notation for the operation of Greedy. Let t be the number of reductions performed by Greedy and let $d_1, d_2, \ldots, d_t$ be the degrees of the vertices selected. The number of vertices removed in the i-th reduction is thus $d_i + 1$. The main property of the algorithm that we use in our analysis is that the sum of the degrees of the vertices removed in the i-th reduction must be at least $d_i(d_i + 1)$. This allows us to lower bound the number of edges removed in each step.

2.1 Previous results on Greedy

The size of any maximal independent set is at least $n/(\Delta+1)$, since for each vertex added to the solution at most Δ others are removed, for a performance ratio of $\Delta + 1$. In fact, a performance ratio of Δ holds, because out of the vertices deleted in each iteration (reduction) at most Δ can belong to the optimal solution.

This can be improved somewhat further by rudimentary arguments. First, observe that Greedy handles isolated and degree one vertices optimally, and that the independence number of a graph with minimum degree two is at most $n\Delta/(\Delta+2)$. Second, if a connected component contains a vertex of degree less than Δ, then Greedy will never remove more than Δ vertices in any step. This then yields a ratio of $\Delta^2/(\Delta + 2) \leq \Delta - 1$. If, however, each vertex is of degree Δ, then the independence number can be seen to be at most $n/2$. That implies a ratio of $(\Delta + 1)/2$, which is at most $\Delta - 1$ for all $\Delta \geq 3$. The above argument is what may have been alluded to in [20, p.306].

In terms of average degree, the celebrated theorem of Turán [21] yields a tight bound on the size of the Greedy solution as well as the independence number. We include here its proof since the proofs of our central results build directly upon it.

Theorem 1 (Turán) $Gr \geq \dfrac{n}{\overline{d}+1}$.

Proof. We count the number of vertices and edges deleted in each reduction of the execution of Greedy.

The removal of vertices in each reduction partitions the vertex set, yielding

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

Since Greedy always picks a vertex of minimum degree, the sum of the degrees of the vertices deleted in step i is at least $d_i(d_i+1)$, and thus the number of edges deleted is at least half that amount. Summing over all the reductions,

$$\frac{\overline{d}}{2}n = |E| \geq \sum_{i=1}^{t}\binom{d_i+1}{2}. \tag{2}$$

We now add (1) and twice (2), and obtain

$$(\overline{d}+1)n \geq \sum_{i=1}^{t}(d_i+1)^2.$$

Using the Cauchy-Schwarz inequality and (1), we get

$$(\overline{d}+1)n \geq n^2/t.$$

Rearranging the inequality, we obtain the desired bound on t, which is precisely the number of vertices found by Greedy. ∎

2.2 Relative size of Greedy solutions

We start by generalizing the constructive version of Turán's theorem by pushing the independence ratio into the expression.

Theorem 2 $Gr \geq \dfrac{1+\tau^2}{\overline{d}+1+\tau}n.$

Proof.

Our proof follows that of Turán's bound, counting the number of vertices and edges deleted in each of the t reductions. This time, we additionally count the number of vertices deleted that belong to some maximum cardinality independent set.

Fix a maximum independent set and let k_i be the number of vertices among the $d_i + 1$ vertices deleted in reduction i that are also contained in that independent set. Then,

$$\sum_{i=1}^{t} k_i = \alpha. \tag{3}$$

Since Greedy always picks a vertex of minimum degree, the sum of the degrees of the vertices deleted in step i is at least $d_i(d_i + 1)$. Note that no edge can have both its end points in the maximum independent set. Then, it can be shown that the number of edges deleted in step i is at least $\binom{d_i+1}{2} + \binom{k_i}{2}$. Hence,

$$\frac{\overline{d}}{2}n = |E| \geq \sum_{i=1}^{t}\binom{d_i+1}{2} + \binom{k_i}{2}. \tag{4}$$

We now add (1), (3) and twice (4), and apply the Cauchy-Schwarz inequality to obtain

$$(\overline{d}+1+\tau)n \geq \sum_{i=1}^{t}(d_i+1)^2 + k_i^2 \geq (1+\tau^2)n^2/t.$$

Rearranging the inequality, we obtain the desired bound on t. ∎

We now turn our attention to bounded-degree graphs, using techniques similar to the preceding proof to obtain bounds parametrized by the maximum degree Δ.

Theorem 3 $Gr \geq \dfrac{1-\tau(1-\tau)}{(1-\tau)\Delta+1}n.$

Proof. The proof follows the proof of the preceding theorem with some extensions. In the i-th step d_i+1 vertices and all edges incident on them are deleted. Of these edges, let x_i have only one end in these d_i+1 vertices; the remaining edges have both ends among the d_i+1 vertices: of these, let y_i of these have one end in the independent set and one outside, and z_i have both ends outside. Then we have

$$x_i + 2(y_i + z_i) \geq d_i(d_i+1), \quad (5)$$

$$y_i \leq k_i(d_i+1-k_i), \quad (6)$$

Multiply (6) by -1 (reversing the inequality) and add it to (5) to obtain

$$\begin{aligned} x_i + y_i + 2z_i &\geq d_i(d_i+1) - k_i(d_i+1-k_i) \\ &\geq \binom{d_i+1}{2} + \binom{d_i+1}{2} - k_i(d_i+1-k_i) \\ &= \binom{d_i+1}{2} + \binom{k_i}{2} + \binom{d_i+1-k_i}{2}. \end{aligned}$$

Since the number of edges deleted in the i-th step is precisely $x_i+y_i+z_i$, we have the following extension of (4):

$$|E| \geq \sum_{i=1}^{t} \binom{d_i+1}{2} + \binom{k_i}{2} + \binom{d_i+1-k_i}{2} - z_i. \quad (7)$$

We also count the total degree of vertices outside the maximum independent set, which entails counting edges incident on the independent set vertices once but those fully outside the independent set twice.

$$(n-\alpha)\Delta \geq \sum_{i=1}^{t} z_i + |E|. \quad (8)$$

Now add twice (7) and twice (8) to obtain

$$2(n-\alpha)\Delta \geq \sum_{i=1}^{t} d_i(d_i+1) + k_i(k_i-1) + (d_i+1-k_i)(d_i-k_i).$$

To simplify the right hand side, we add $\sum_{i=1}^{t} d_i + k_i + (d_i - k_i)$ and compensate for this by adding $2n$ to the left hand side (invoking (1)). Then,

$$2\Delta(n-\alpha)+2n \geq \sum_{i=1}^{t}(d_i+1)^2 + k_i^2 + (d_i+1-k_i)^2. \quad (9)$$

Using (1), (3) and the Cauchy-Schwarz inequality, we obtain

$$(2\Delta(1-\tau)+2\tau)n \geq [1+\tau^2+(1-\tau)^2]n^2/t.$$

The claim follows from this. ∎

2.3 Performance guarantee

The following bound on the performance of Greedy on sparse graphs follows immediately since the ratio of the independence number τn to the bound in thm. 2 is maximized when $\tau = 1$.

Corollary 4 $\rho_{Gr} \leq \dfrac{\overline{d}+2}{2}.$

For bounded-degree graphs, the general expression obtained in thm. 3 almost – but not quite – yields our main claim about the performance ratio of Greedy. We now proceed to analyze the performance of Greedy using a finer scalpel.

Let $t_{d,k}$ be the number of reductions performed by Greedy where a vertex of degree d was chosen and exactly k vertices of the independent set were removed. More precisely, for $d = 0, 1, \ldots, \Delta$ and $k = 0, 1, \ldots, d$, we define $t_{d,k} = |\{i : d_i = d \text{ and } k_i = k\}|$. With this notation, we may rewrite the constraints (1), (3) and (9) as

$$\sum_{d,k}(d+1)t_{d,k} = n, \quad (10)$$

$$\sum_{d,k} k t_{d,k} = \alpha, \quad (11)$$

$$\sum_{d,k}[(d+1)^2 + k^2 + (d+1-k)^2]t_{d,k} \leq 2n(\Delta+1) - 2\Delta\alpha. \quad (12)$$

We wish to extract from these constraints the best possible lower bound for $t = \sum_{d,k} t_{d,k}$. For this we use the method of multipliers described in Chvátal [6, page 54] (see Chvátal [5] for an elegant application of this method to analyze the greedy heuristic for the set covering problem.)

Theorem 5 $\rho_{Gr} \leq (\Delta+2)/3.$

Proof. We need to consider two cases based on the value of Δ.

Case $\Delta \equiv 0, 1 \pmod 3$ [i.e. $\Delta+1 \equiv \pm 1 \pmod 3$]. We construct a linear combination of constraints (10) and (12), with multipliers $2(\Delta+1)$ and -1 respectively:

$$\sum_{d,k} C(d,k)t_{d,k} \geq 2\Delta\alpha$$

where

$$C(d,k) \;=\; 2(d+1)(\Delta+1)-((d+1)^2+k^2+(d+1-k)^2).$$

We show below that

$$\max_{d,k} C(d,k) \;=\; 2\Delta(\Delta+2)/3. \qquad (13)$$

Hence,

$$\frac{2}{3}\Delta(\Delta+2)\sum_d t_d \;\geq\; 2\Delta\alpha,$$

that is,

$$\frac{\Delta+2}{3}t \;\geq\; \alpha,$$

as required.

It remains to establish (13). Let $f_0 : \Re \to \Re$ and $f_1 : \Re \to \Re$ be defined by

$$f_0(x) = 2x(\Delta+1) - \frac{3}{2}x^2, \quad f_1(x) = f_0(x) - \frac{1}{2}.$$

It can be easily verified that

$$\max_k C(d,k) = \begin{cases} f_0(d+1) & \text{if } d \text{ is odd} \\ f_1(d+1) & \text{if } d \text{ is even} \end{cases}$$

Now, $f_0'(x), f_1'(x) = 0$ iff $x = 2(\Delta+1)/3$, and $f_0''(x), f_1''(x) = -3$. Thus, both f_0 and f_1 are concave functions that achieve their unique maximum at $\hat{x} = 2(\Delta+1)/3$. Since f_0 and f_1 are polynomials of degree 2 in x, $f_0(\hat{x}+\epsilon) = f_0(\hat{x}-\epsilon)$ and $f_1(\hat{x}+\epsilon) = f_1(\hat{x}-\epsilon)$, for all ϵ. Thus, to establish the claim, it is enough to verify that f_0 at the nearest even integer to $\hat{x}$ and f_1 at the nearest odd integer to $\hat{x}$ are at most $2\Delta(\Delta+2)/3$.

Let $\Delta+1 = 3m+r$, where $r = \pm 1$ (since $\Delta+1 = \pm 1 \pmod 3$). The nearest even integer to $\hat{x}$ is $2m$ and the nearest odd integer to $\hat{x}$ is $2m+r$. Plugging in, we find that

$$f_0(2m) \;=\; f_1(2m+r) \;=\; 6m^2+4mr \;=\; \frac{2}{3}\Delta(\Delta+2)$$

establishing the claim.

Case $\Delta \equiv -1 \pmod 3$. This time, we construct a linear combination of the constraints (10), (11) and (12), with multipliers $2(\Delta+1)$, 2 and -1, respectively:

$$\sum_{d,k} C(d,k)t_{d,k} \;\geq\; 2(\Delta+1)\alpha,$$

where

$$C(d,k) = 2(d+1)(\Delta+1) + \beta(d,k)$$

and

$$\beta(d,k) = 2k(d+1)^2 - k^2 + (d+1-k)^2 = 2k(d+2-k).$$

Note that

$$\max_k \beta(d,k) = \frac{1}{2}(d+2)^2.$$

We want an upper bound only in terms of Δ of the function $f : \Re \to \Re$, defined by

$$f(x) = 2(x+1)(\Delta-x) + \frac{1}{2}(x+2)^2,$$

for x integral. Indeed, $f'(x) = 0$ iff $x = 2\Delta/3$ and $f''(x) = -3$; thus f has its unique maximum at $\hat{x} = 2\Delta/3$. Since f is a polynomial of degree 2 in x, it is enough to establish the bound at the integer nearest to $\hat{x}$, namely $2(\Delta+1)/3$. By inspection we find that

$$C(d,k) \;\leq\; f(\frac{2(\Delta+1)}{3}) \;=\; \frac{2}{3}(\Delta+1)(\Delta+2)$$

from which the claim follows. ∎

2.4 Limitations

The performance ratios proved above cannot be improved.

Theorem 6 $\rho_{Gr} \;\geq\; \dfrac{\Delta+2}{3} - O(\Delta^2/n)$, *for* $\Delta \geq 3$.

Proof. We give a detailed construction for $\Delta \equiv 1 \pmod 3$. We construct a graph, parametrized by integer $l \geq 2$, that consists of a chain of repetitions of a pair of subgraphs: a clique on l vertices followed by an independent set on l vertices. The two subgraphs are completely connected, while the connections between the independent set and the clique of the following pair miss only a single matching (i.e. each vertex is of degree $l-1$). The chain ends with one additional clique.

An instance of this graph with $l = 3$ is shown in fig. 1, with the vertices picked by Greedy shown in black and the maximum independent set vertices in grey.

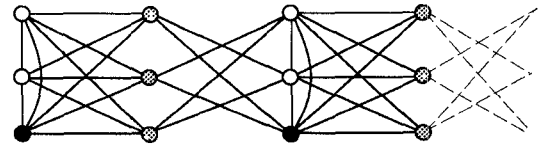

Figure 1: Greedy performance lower bound example

The essential property of the graph is that the degree of the independent set vertices equals the degree of the vertices of the first clique of the chain. We can therefore assume that Greedy will pick one of the vertices from the first clique and remove the remaining vertices from the pair, reducing the graph to an identical chain with one fewer pairs. Thus, Greedy selects one vertex from each pair, plus one from the final clique, for a total of

$(n-l)/2l+1$. The optimal solution contains all the independent set vertices for a total of $(n-l)/2$. This yields a ratio of

$$\rho_{Gr} \geq l - 2l^2/n.$$

To relate that to the degree measures, we have that $\Delta = 3l-2$, and

$$\begin{aligned}\overline{d} &\leq 2\left(\binom{l}{2} + l^2 + l(l-1)\right)/2l \\ &= \frac{5l-3}{2}.\end{aligned}$$

Thus,

$$\rho_{Gr} \geq \frac{\Delta+2}{3} - O(\Delta^2/n), \tag{14}$$

and

$$\rho_{Gr} \geq \frac{2\overline{d}+3}{5} - O(\overline{d}^2/n), \tag{15}$$

even when $\tau \leq 1/2$.

For values of $\Delta \equiv 0, 2 \pmod 3$ we need chains of more complicated groups of six subgraphs. For 0 (mod 3), the groups are of the form

$$K_{l-1} \; -- \; \overline{K_l} \; -- \; K_{l-1} \; -- \; \overline{K_l} \; -- \; K_l \; -- \; \overline{K_{l-1}}$$

In all cases, a clique is completely connected to the following independent set, while connections from the independent set to the next clique miss a single matching. In addition, $l-1$ edges from the second subgraph (the first independent set) go towards the first subgraph in the next group in the chain rather than to the next subgraph in the current group. The chain is finished with an additional clique as before.

An example for $l = 3$ is given in fig. 2.

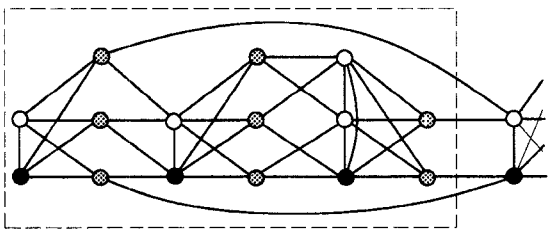

Figure 2: Hard graph for Greedy when $\Delta = 6$

The graph is carefully designed so that at each point in time the minimum degree will be $2l-2$, namely the degree of the left-most remaining clique. Hence, we may assume that Greedy will pick exactly one vertex per clique, or 3 nodes per group, while the number of vertices deleted from the largest independent is $3l-1$ and the maximum degree is $3l-3$. Hence, ignoring the end of the chain, the performance ratio attained is $l - 1/3 = (\Delta+2)/3$.

The case of $\Delta \equiv 2 \pmod 3$ is similar, with each group of the form

$$K_{l-1} \; -- \; \overline{K_{l+1}} \; -- \; K_l \; -- \; \overline{K_l} \; -- \; K_l \; -- \; \overline{K_l}$$

and edges going from the second to the fifth subgraph. We leave the details to the curious reader. ∎

We can also show that the bound on the performance ratio in terms of average degree (Corollary 4) approaches optimality as $\overline{d}$ gets larger.

Theorem 7 $\rho_{Gr} \geq \frac{\overline{d}+2}{2} - O(1/\overline{d})$.

The graph for which this ratio is attained consists of a chain of pairs of subgraphs as in the previous example, with each clique reduced to a single vertex adjacent to several of the following single-vertex cliques. We omit the detailed proof.

Variations of the above constructions show that the bounds of thms. 2 and 3 are tight for a range of values of τ. This involves varying the size of the cliques relative to the size of the independent sets, possibly by connecting each independent set to several subsequent cliques. We omit the details.

2.5 Nemhauser-Trotter

A method of Nemhauser and Trotter [15] for the fractional vertex cover problem has been used successfully to obtain better approximations for both the vertex cover and the independent set problems on bounded-degree graphs. The time complexity of this method is $O(n^{3/2} + |E|)$.

Their method yields an optimal solution to the linear relaxation of the integer problem where each variable, representing whether a node is in the independent set or the vertex cover, has value from $\{0, 1, 1/2\}$. This corresponds to partitioning the vertices into three sets R, P, and Q with the following important properties:

1. Some maximum independent set of G contains all the vertices of R and none of the vertices in P.
2. The independence fraction of the subgraph H induced by Q is at most $1/2$.

This in effect means that the method can be used as a preprocessor for approximation algorithms; since the portions R and P are solved optimally, it suffices to focus our attention on the graph H where the independence number is guaranteed to be small.

This immediately implies, for instance, that the $n/(\Delta+1)$ lower bound on the size of the Greedy solution along with the above upper bound on the independence number yields a $(\Delta+1)/2$ performance ratio

for the combination of Greedy and Nemhauser-Trotter methods. A stronger bound follows immediately from thm. 3.

Corollary 8 $\rho_{Gr+NT} \leq \frac{\Delta+2}{3}$.

It follows from (14) that this ratio is essentially tight for $\Delta \equiv 1 \pmod 3$. That is, if the only property used about the Nemhauser-Trotter method is that it allows us to assume that the independence fraction is at most half, then we can do no better than we have shown. As we have not analyzed the method in detail, it is possible however that it will have other properties that inhibit examples where Greedy has poor performance.

We can improve on this bound slightly to $(\Delta+2)/3 - 1/(3\Delta+2)$ when $\Delta \equiv 0, 2 \pmod 3$. For instance, we have a tight ratio of $3/2$ when $\Delta = 3$. Also, for $\Delta = 5$ we get a ratio of $16/7 \approx 2.286$, down from $2.3\overline{3}$, while there is a graph that forces a ratio of 2.27. We omit the details.

2.6 Average degree

Once Nemhauser-Trotter has been applied, the average degree may have changed for the worse and we cannot immediately apply the bounds proved on Greedy. Nevertheless, a closer look shows that the bounds will complement each other as hoped for. Hochbaum [11], showed that the Turán bound on Greedy can be complemented with the $\tau \leq 1/2$ promise of Nemhauser-Trotter to yield a performance ratio of $(\overline{d}+1)/2$.

Our bound on Greedy, when complemented by Nemhauser-Trotter, yields a considerably performance ratio. The proof is similar to that of Hochbaum and is omitted.

Theorem 9 $\rho_{Gr+NT} \leq \frac{2\overline{d}+3}{5}$.

Proof. The proof follows Hochbaum. Let $\overline{d}'$, τ', n' denote the average degree, independence fraction, and number of vertices of H, respectively. We also denote the size of the vertex sets R and P simply by their names.

We shall be needing two properties. The first is that $R \geq P$, as otherwise G itself will have the desired properties. The second is that the number of edges in G are at least $\overline{d}'n' + R + P$, since we may assume that G is a connected graph. Combined, this means that

$$\overline{d} \geq \frac{\overline{d}'n' + 2R}{n' + 2R}.$$

The size of the greedy solution will be $R + \frac{1+\tau'^2}{\overline{d}'+1+\tau'}n'$, where n' and $\overline{d}'$ are the number of vertices and average degree of H, respectively. The size of the optimal solution is $R + \frac{1}{2}n'$. The claim now reduces to showing that

$$\frac{R + n'/2}{R + \frac{1+\tau'^2}{\overline{d}'+1+\tau'}n'} \leq \frac{2}{5}\frac{\overline{d}'n' + 2R}{n' + 2R} + \frac{3}{5}.$$

It can be easily observed that, as before, the left hand side is maximized when τ' is at its maximum. Hence, we plug $\tau' = 1/2$, merge the two terms on the right hand side, cross-multiply, and simplify to obtain

$$2 \leq \frac{5}{2\overline{d}'+3} + \frac{2\overline{d}'+3}{5},$$

which is true since $a/b + b/a$ is always at least 2. ∎

This again is tight for $\Delta \equiv 1 \pmod 3$, from (15).

3 Subgraph Removal

We present a generally applicable 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 comes from the observation that graphs without dense subgraphs, particularly cliques, contain provably 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 and our performance ratio will improve in either case. This idea was previously used to approximate the Independent Set problem in general graphs [4].

This schema uses as subroutine two types of algorithms: an approximation algorithm for general graphs, and algorithms that find large independent sets in k-clique free graphs with possibly different algorithms for different values of k. Except for the case of triangle-free graphs, we consider here only the case of Greedy. Better algorithms of either type translate immediately to better performance ratios. In fact, we have recently improved the best performance ratios known for moderate to large values of Δ [10] using better algorithms for clique-free graphs.

We first illustrate this technique in its simplest form, when removing disjoint 2-cliques (i.e. a matching). The next step is to consider removing triangles, where we can use an effective algorithm of Shearer. We then present the general schema, and prove improved ratios for Greedy in the context of this scheme.

Removing edges The following simple idea has also been considered in [17] and [3]. We use it to get a linear time algorithm with a good performance ratio in terms of average degree.

Find an independent in two ways, and retain the larger one. One way is to find a maximal matching with m edges, and use the complement – the vertices not appearing in the matching – as an independent set approximation. The size of this set is $n - 2m$ which is at least $2\alpha - n$, since the independence number α is at most $n - m$. Hence, the performance ratio is at most $\frac{\tau}{2\tau-1}$. The other way is to use the Greedy algorithm, with a performance guarantee of $\frac{(\overline{d}+1+\tau)\tau}{1+\tau^2}$ (see thm. 4).

Observe that the former ratio is monotone decreasing with τ ($\tau > 1/2$), while the latter one 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, this yields a performance ratio of $\frac{2\overline{d}+4.5}{5}$. Hence, in fully linear time we come within an additive 0.3 of the $(2\overline{d}+3)/5$ bound in sec. 2.6 that requires $\Omega(n^{3/2})$ time.

Removing triangles This 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 3-cliques, the remaining graph will not be independent, yet will be more amenable to the discovery of large independent subgraphs.

Based on a result of Ajtai, Komlós, and Szemerédi [1], Shearer [18, 19] gave an efficient algorithm that finds an independent set of size $\Omega(n \log \overline{d}/\overline{d}) \geq \Omega(n \log \Delta/\Delta)$ in a triangle-free graph. If a fixed constant fraction of the input graph is triangle-free, Shearer's algorithm yields a performance ratio of $O(\Delta/\log \Delta)$; otherwise, the independence fraction must be close to one third, in which case Greedy yields a performance ratio of $\Delta/3.5 + O(1)$.

The general schema We now present the general clique removal schema. From the input graph, it produces graphs free of induced ℓ-cliques, for all ℓ between 2 and the parameter k. On each of these graphs, including the input graph, it runs the appropriate independent set algorithm and outputs the largest of those solutions.

In the case considered here, Greedy is used as the approximation algorithm on general graphs in the first line, as well as the algorithm for ℓ-clique-free graphs, $\ell \geq 4$, while Shearer is used for triangle-free graphs.

```
SubgraphRemoval(G,k)
  X_∞ ← Greedy(G)
  for ℓ = k downto 2 do
    S ← Maximal collection of disjoint ℓ cliques in G
    G ← G − S
    case ℓ do
      ℓ ≥ 4 : X_ℓ ← Greedy(G)
      ℓ = 3 : X_ℓ ← Shearer(G)
      ℓ = 2 : X_ℓ ← V(G)
    od
  od
  Output X_i of maximum cardinality
end
```

The time complexity of the algorithm remains nearly linear for constant-degree graphs. A maximal collection of disjoint k-cliques can be found in time $|E|\binom{\Delta}{k-2} \leq n\binom{\Delta}{k-1}$, and the Greedy algorithm can be implemented in linear time. Shearer's algorithm runs in randomized linear time, or in $O(\min(\Delta^2 n, n \log n) + |E|)$ otherwise.

Improving the performance of Greedy We shall now analyze to some extent the performance of the algorithm above for $k \geq 4$. In the light of the improved ratios recently obtained by replacing Greedy by other methods, the following theorem is given primarily as a "proof of concept".

First, we observe that the lack of k-cliques does improve the size of the solution output by Greedy.

Lemma 10 *Let G be a graph containing no clique on $\ell \geq 4$ vertices. For any subgraph H of G,*

$$Gr(H) \geq \frac{1 + \tau(H)^2 + (1 - \tau(H))^2/(\ell - 3)}{\Delta + 1 + \tau(H) + 2(1 - \tau(H))/(\ell - 3)} n(H).$$

The proof is an easy modification of the proof of Turán's bound, where the lack of $r - 1$ cliques in the neighborhood of the selected vertex allows us to declare that at least $d_i^2 - d_i^2 \frac{r-3}{2(r-2)}$ edges are deleted in the i-th reduction.

Theorem 11 *The performance ratio of* SubgraphRemoval(4) *is at most $\Delta/3.67 + c$, where c is a constant independent of Δ.*

Proof. Let n_4 denote the size of the 4-clique free subgraph found. Note that the size of the 3-clique free subgraph is $o(n)$ as otherwise Shearer would yield an asymptotically better solution. The independence fraction of the 4-clique free subgraph is therefore 1/3. Hence, Greedy on the that subgraph yields a solution of asymptotically $(14/9)/\Delta n_4$. Let ρ denote the coefficient in front of Δ in the performance ratio obtained by SubgraphRemoval(4). Then, $\rho \leq (9/14)\tau n/n_4$, and $n_4 \leq (9/14)(\tau/\rho)n$. Note that $\tau \leq 1/4 + (1/12)n_4/n$. Combining the two, $\tau \leq 1/4 + (3/56)\tau/\rho$, which simplifies to $\tau \leq 1/(4(1 - 3/(56\rho)))$. If τ is greater than 0.311 then $\rho \leq 1/3.67$. On the other hand, if τ is smaller, Greedy in the input graph yields the desired performance ratio, by thm. 3. ∎

The best ratio we have obtained with technique on Greedy is approximately $\Delta/3.76$ by removing 8-cliques. The above performance ratio is an asymptotic value for large degrees, but the Subgraph Removal method also does help for small degree. In fact, it yields an improvement over Greedy for every $\Delta \geq 6$. In this case, we need to replace SubgraphRemoval(2) with the more effective Nemhauser-Trotter procedure. For instance, for $\Delta = 6$ it improves the performance ratio from $2.6\overline{6}$ to 2.56.

Vertex Cover Approximation The combination of Greedy and Nemhauser-Trotter can be used as an effective vertex cover approximation algorithm. Hochbaum proved a bound of $2 - 2/(\Delta + 1)$. Our results improve this ratio to $2 - 3/(\Delta + 2)$.

The method of Shearer improves on the previous bounds given for triangle-free graphs and used by Monien and Speckenmeyer [14] to obtain better vertex cover approximations in combination with triangle-removal. If $f_{Sh}(\Delta)n$ denotes the bound proved for Shearer's algorithm in [19], then the performance ratio obtained is at most $2 - f_{Sh}(\Delta)$. Given that f_{Sh} is an asymptotic improvements over previous bounds, we observe the following interesting corollary.

Corollary 12 $\rho^{VC} \leq 2 - \frac{\log \Delta + O(1)}{\Delta}$.

4 Parallel and Distributed Algorithm

The Greedy algorithm stipulates that in each step a vertex of globally minimum degree be selected, added to the solution, and removed from the graph along with its neighbors. As such, it looks impossible to parallelize, as well as offering little freedom for heuristic improvements. Fortunately, this is one of the delightful cases when the analysis guides us towards the design of better and/or more general algorithms.

We observe that the selection of a vertex is prescribed by a simple local rule and that a constant fraction of the vertices in a bounded-degree graph satisfies this rule. This has some interesting implications. For one, it opens up the possibility of the design of heuristics using secondary selection rules that retain the performance ratios of thms. 2 and 3. Another is a straightforward derivation of a parallel as well as a distributed approximation algorithm attaining these performance ratios.

From the proofs of thms. 2 and 3, we find that a sufficient criteria for the selected vertex v is that its degree be less than the average of its neigbors' degrees. That is,

$$d(v) \leq \frac{\sum_{w \in N(v)} d(w)}{d(v)}. \qquad (16)$$

A heuristic may choose any ordering that obeys the above property. Vertices can be selected in parallel as long as the selection of one doesn't affect the above criteria for the other. In particular, vertices with disjoint and non-adjacent neighborhoods (i.e. of distance three or greater) can be selected and processed concurrently.

This suggests a natural approach to a parallel algorithm:

1. Find a set W of vertices satisfying (16).
2. Form a graph H on vertex set W, with edge between vertices that may interact with respect to the Greedy reduction (i.e. of distance three or less).
3. Find a maximal independent set MIS in H.
4. Perform the Greedy reduction on these vertices in parallel. Namely, add the vertices MIS to the solution, and delete them and their neighbors from the graph.
5. Repeat from step 1 until the graph is empty.

The following lemma due to Alon and Szegedy (private communication) shows that a significant fraction of the vertices must have the above property simultaneously.

Lemma 13 *In a graph on n vertices with maximum degree Δ, at least $\frac{4}{\Delta^2+4}n$ vertices satisfy property (16).*

Proof. Let D_v denote $d(v)^2 - \sum_{w \in N(v)} d(w)$. We shall show that

$$\Pr_v[D_v \geq 0] \geq \frac{4}{\Delta^2 + 4},$$

which implies the lemma. As observed by Shearer [18], $E_v[D_v] = 0$. The value of D_v is bounded above by $d(v)(\Delta - d(v)) \leq \Delta^2/4$, and since it is integral, it must differ from zero by at least one when negative. Thus,

$$-1(1 - \Pr[D_v \geq 0]) + \Delta^2/4 \cdot \Pr[D_v \geq 0] \geq 0.$$

The claim now follows. ∎

Theorem 14 *There is an EREW parallel algorithm that finds an independent set of size and performance satisfying theorems 3 and 5 in time $\log^* n \min(poly(\Delta) \log n, \Delta^{\Delta})$ using n processors.*

Proof. Each vertex added to the solution will satisfy property (16) regardless of the order of removal of the simultaneously chosen vertices. Hence, the results of the theorems apply to this algorithm.

Let us now estimate the time complexity, starting with the number of iterations. The first step reduces the number of vertices by a factor of at most $O(\Delta^2)$, as per the lemma above. The number of vertices deleted in the fourth step, which are the selected vertices and their neighbors, is at most another Δ^2 factor smaller. Thus at

least n/Δ^4 vertices are removed from the graph in each step, so the number of rounds is bounded by $\Delta^4 \log n$. Also notice that for any vertex in the graph, some vertex of distance at most Δ gets eliminated in each round. Hence the number of rounds is also bounded by Δ!

The only non-trivial step in each round is the computation of a maximal independent set (MIS) of the graph H. An algorithm of Goldberg et al. [8] finds an MIS in time $O(\Delta(H)\log\Delta(H)(\Delta(H)+\log^* n))$ using linear number of processors. The combined time complexity is therefore bounded by $O(\Delta^7 \log\Delta(\Delta^3+\log^* n)\log n)$. The processor count is linear in n, and considering the total amount of work can probably be made some polynomial of Δ smaller. ∎

The algorithm given above also satisfies the criteria of a distributed algorithm.

Acknowledgments We thank Noga Alon and Mario Szegedy for kindly proving lemma 13. We also thank a referee for suggesting that the parallel algorithm might imply a distributed one.

References

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

[2] N. Alon and J. Spencer. *The Probabilistic Method.* Wiley, 1992.

[3] P. Berman and M. Fürer. Approximating maximum independent set in bounded degree graphs. In *Proc. Fifth ACM-SIAM Symp. on Discrete Algorithms*, Jan. 1994.

[4] R. B. Boppana and M. M. Halldórsson. Approximating maximum independent sets by excluding subgraphs. *BIT*, 32(2):180–196, June 1992.

[5] V. Chvátal. A greedy heuristic for the set covering problem. *Math. Oper. Res.*, 4:233–235, 1979.

[6] V. Chvátal. *Linear Programming.* Freeman, New York, 1983.

[7] P. Erdős. On the graph theorem of Turán (in Hungarian). *Mat. Lapok*, 21:249–251, 1970.

[8] A. V. Goldberg, S. A. Plotkin, and G. E. Shannon. Parallel symmetry-breaking in sparse graphs. In *Proc. 19th ACM Symp. on Theory of Computing*, pages 315–324, May 1987.

[9] M. Goldberg and T. Spencer. An efficient parallel algorithm that finds independent sets of guaranteed size. In *Proc. First ACM-SIAM Symp. on Discrete Algorithms*, pages 219–225, Jan. 1990.

[10] M. M. Halldórsson and J. Radhakrishnan. Improved approximations of independent sets in bounded-degree graphs. Manuscript, Feb. 1994.

[11] D. S. Hochbaum. Efficient bounds for the stable set, vertex cover, and set packing problems. *Disc. Applied Math.*, 6:243–254, 1983.

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

[13] S. Khanna, R. Motwani, M. Sudan, and U. Vazirani. On syntactic versus computation views of approximability. Manuscript, Dec. 1993.

[14] B. Monien and E. Speckenmeyer. Some further approximation algorithms for the vertex cover problem. In *Lecture Notes in Computer Science #159*, pages 341–349. Springer Verlag, 1983.

[15] G. L. Nemhauser and W. T. Trotter. Vertex packing: Structural properties and algorithms. *Math. Programming*, 8:232–248, 1975.

[16] A. Panconesi and A. Srinivasan. Improved distributed algorithms for coloring and network decomposition problems. In *Proc. 24nd Ann. ACM Symp. on Theory of Computing*, pages 581–592, May 1992.

[17] V. T. Paschos. A $(\delta/2)$-approximation algorithm for the maximum independent set problem. *Inf. Process. Lett.*, 44:11–13, Nov. 1992.

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

[19] J. B. Shearer. A note on the independence number of triangle-free graphs, II. *J. Combin. Theory Ser. B*, 53:300–307, 1991.

[20] H. U. Simon. On approximate solutions for combinatorial optimization problems. *SIAM J. Disc. Math.*, 3(2):294–310, May 1990.

[21] P. Turán. On an extremal problem in graph theory (in Hungarian). *Mat. Fiz. Lapok*, 48:436–452, 1941.