An Approximation Algorithm for Bounded Degree Deletion*

Tomáš Ebenlendr[†] Petr Kolman[‡] Jiří Sgall[‡]

Abstract

Bounded Degree Deletion is the following generalization of Vertex Cover. Given an undirected graph $G = (V, E)$ and an integer $d \geq 0$, what is the minimum number of nodes such that after deleting these nodes from G, the maximum degree in the remaining graph is at most d?

We give a randomized $O(\log d)$-approximation algorithm for this problem. We also give a tight bound of $\Theta(\log d)$ on the integrality gap of the linear programming relaxation used in our algorithm. The algorithm is based on the algorithmic version of the Lovász Local Lemma.

1 Introduction

In the Bounded Degree Deletion problem we are given an undirected graph $G = (V, E)$ and an integer $d \geq 0$. A feasible output is a set of nodes Z such that the graph induced by $V - Z$ has the maximal degree at most d. The objective is to minimize $|Z|$, the number of removed nodes.

The special case of Bounded Degree Deletion for $d = 0$ is Vertex Cover. This implies that Bounded Degree Deletion is APX-hard and the best possible approximation ratio is a constant.

In this note we focus on the asymptotic dependence of the approximation ratio on d. We present a randomized $O(\log d)$-approximation algorithm for Bounded Degree Deletion.

*Partially supported by Institutional Research Plan No. AV0Z10190503, Inst. for Theor. Comp. Sci., Prague (project 1M0545 of MŠMT ČR), and grant IAA100190902 of GA AV ČR.

[†]Institute of Mathematics, AS CR, Žitná 25, CZ-11567 Praha 1, Czech Republic. Email: ebik@math.cas.cz.

[‡]Dept. of Applied Mathematics, Faculty of Mathematics and Physics, Charles University, Malostranské nám. 25, CZ-11800 Praha 1, Czech Republic. Email: kolman,sgall@kam.mff.cuni.cz.

Bounded Degree Deletion is a special case of Hitting Set: the sets to hit are those that correspond to vertex-sets of stars with $d+1$ leaves. It is easy to give a $(d+2)$-approximation algorithm. Formulate a linear program which asserts that from each star with $d+1$ leaves at least one node is selected. Then, given a fractional solution, choose all the nodes whose variable is at least $1/(d+2)$. Unfortunately, the integrality gap of the linear program is $d+2$, thus, it cannot yield a better approximation. The exact formulation and discussion are presented in the next section.

To get another approximation, one can cast the problem as an instance of the so-called covering integer programs [5]. Again, we formulate a linear program for Bounded Degree Deletion; this time it contains at most one constraint for each maximal star. In particular, for each vertex there is at most one constraint and the constraint for a vertex of degree $\delta > d$ asserts that either the vertex is chosen or at least $\delta - d$ of its neighbors are chosen. The general framework of Kolliopoulos and Young yields $O(\log n)$-approximation algorithm.

To obtain the better $O(\log d)$-approximation algorithm, we use yet another linear programming relaxation. For each star with $d+i$ leaves, the corresponding constraint asserts that either the center is chosen or at least i leaves are chosen. Rounding the fractional solution is more difficult here since the coefficients of the linear program are proportional to the degree of nodes. As our main tool, we use the algorithmic version of the Lovász Local Lemma in the formulation of Srinivasan [9]. The details are given in Section 3.

As our second result, in Section 4, we show that the integrality gap of our linear program (2) is $\Theta(\log d)$. Thus this formulation cannot yield a better approximation factor (asymptotically in d) and our approximation algorithm is in this sense optimal.

It remains an open problem whether there exists a c-approximation algorithm for Bounded Degree Deletion such that c is a constant independent of d.

Previous work and motivation

Bounded Degree Deletion was previously studied mainly in the context of fixed parameter tractability also under the name Almost d-Bounded Graph [4, 7]. There the goal is, generally speaking, to design exact algorithms that work, for graphs of size n with the optimal solution of size k, in time $poly(n) \cdot f(k)$ for some possibly exponential function f, or to prove related hardness results.

Fixed parameter tractability is also studied for the complementary problem of finding maximal s-plexes [8, 1, 6]. Here s-plex is an induced subgraph on n' nodes such that the minimal degree is at least $n' - s$. Thus finding maximal s-plexes is in the same relation to Bounded Degree Deletion as is Maximal Clique to Vertex Cover.

Both of these problems are motivated by processing data with errors in the context of biological applications and analysis of social networks. Generally, the graph edges denote (in)consistencies between different nodes and the goal is to remove the smallest possible number of data points so that the remaining ones are consistent (or to find the biggest consistent subset). Due to the noise in data, it may be unrealistic to expect to see a perfect clique (or an independent set). Removing fewer data points at the cost of a limited inconsistency may be a more realistic approximation for the given application. Bounded Degree Deletion and s-plex maximization are possible formalizations of this approach. We refer to papers [7, 3, 2, 1, 8] and references therein for further discussion of these motivations.

We are not aware of any previous study of approximation algorithms for Bounded Degree Deletion, besides the straightforward application of algorithms for Hitting Set and covering integer programs, as described above.

2 Preliminaries and the linear programming formulations

A *star* is a tree consisting of a node v and a set L of nodes connected to v; the node v is the *center* of the star and the nodes in L are the *leaves*. It will be convenient to denote a star with a center u and leaves L as an ordered pair (u, L). We use the term k-*star* to denote a star with k leaves. Given a graph $G = (V, E)$, let $\mathcal{S}_k$ denote the set of all k-stars that are subgraphs of G (not necessarily induced).

For each $v \in V$ and $A \subseteq V$ we define $N^A(v) = \{u \in A \mid \{u, v\} \in E\}$, the set of neighbors of v in A, and $\delta^A(v) = |N^A(v)|$, the degree of v relative to A.

Bounded Degree Deletion can be treated as a special case of Hitting Setwhere the sets to be hit are all $(d + 1)$-stars. This leads to the following linear programming relaxation.

$$
\begin{aligned}
& & \min \sum_{u \in V} x_u & & \\
\forall (u, L) \in \mathcal{S}_{d+1} : & & x_u + \sum_{v \in L} x_v & \geq 1 & \quad (1)\\
& & 0 \leq x_u \leq 1 & &
\end{aligned}
$$

It is easy to see that integral solutions of (1) exactly correspond to feasible outputs of Bounded Degree Deletion and the objective is the same as well. Given a fractional solution $x_v, v \in V$, we take Z to be the set of all v with $x_v \geq 1/(d+2)$. This gives a $(d+2)$-approximation as at least one node in each $(d+1)$-star is chosen into Z and the size of Z is bounded by $|Z| \leq (d+2) \sum_{v \in V} x_v$. However, $d+2$ is also the integrality gap of the linear program of (1) as we will show. Consider the complete graph on n nodes. Then the integral optimum is $n - d - 1$ while the fractional optimum has $x_v = 1/(d+2)$ for all $v \in V$ and the objective is $\sum_{v \in V} x_v = n/(d+2)$. For n large, the gap approaches $d + 2$.

To overcome this barrier, we add additional inequalities for larger stars. We use the following stronger relaxation.

$$
\begin{aligned}
& & \min \sum_{u \in V} x_u & & \\
\forall i \geq 1\,, \quad \forall (u, L) \in \mathcal{S}_{d+i} : & & i \cdot x_u + \sum_{v \in L} x_v & \geq i & \quad (2)\\
& & 0 \leq x_u \leq 1 & &
\end{aligned}
$$

We pause to note that the number of inequalities in the linear program may be exponential in the size of G, nevertheless, the linear program is solvable in polynomial time using the ellipsoid method (given a vector x, one can easily check whether there is a violated inequality).

Clearly, the integral solutions of (2) again correspond to feasible outputs of Bounded Degree Deletion. The constraint for $(u, L) \in \mathcal{S}_{d+i}$ is satisfied exactly if either the center is in Z or if at least i leaves are in Z.

For integral solutions, we can take only the constraints for stars $(u, N^V(u))$, that is, the maximal star for each node. The set of the feasible integral solutions remains unchanged and the number of constraints is only linear. However, we do not know what is the integrality gap of the corresponding linear program.

3 The approximation algorithm

In this section we present the $O(\log d)$-approximation algorithm. Our approach is as follows. We take an optimal solution x_u, $u \in V$ of the linear program (2). Then we choose each node u into Z independently at random with probability αx_u where α is roughly the desired approximation factor. To show that all the degrees are at most d after removing Z, with non-zero probability, we use the Lovász Local Lemma. To finally obtain the algorithm, we use the algorithmic version of the Lovász Local Lemma, in the framework of Srinivasan [9]. Due to the use of the Lovász Local Lemma, we cannot argue that the number of chosen nodes is small directly from Chernoff bounds, as usual. Instead, we argue about the number of chosen nodes in smaller groups of $O(d)$ nodes, to limit the dependency of the bad events. To make this work, we further modify the random sampling so that nodes with small x_u (proportional to $1/d$) are never chosen.

Before we continue with the formal proof, we state the constructive version of the Lovász Local Lemma according to Srinivasan [9, Theorem 2.1]. The notation is slightly changed for our purposes and the particular constant 0.05 is chosen according to [9, Remark 2.1] asserting that any constant strictly smaller than $4/(27e)$ is sufficient. (The particular constant is irrelevant for our construction.)

Theorem 3.1 ([9]) *Suppose that we have independent random variables X_v, $v \in W$, and a collection of polynomially many "bad" events $E_1, \ldots, E_m$ with support sets $T_1, \ldots, T_m \subseteq W$ such that the each event E_i is completely defined from variables $(X_v \mid v \in T_i)$. Let D be an integer. Assume that the following conditions hold:*

(S1) For every i, there are at most D values of j, $j \neq i$, such that $T_i \cap T_j \neq \emptyset$.

(S2) We have $pD^4 \leq 0.05$, where $p = \max_i \mathbf{Pr}[E_i]$.

(S3) Each X_v can be sampled in randomized polynomial time and the truth value of each E_i can be determined from the values of $(X_i, i \in T_i)$ in polynomial time.

Then there exists a randomized polynomial time algorithm that with high probability finds an assignment of X_i which avoids all events E_i.

Now we define our random variables, bad events and continue towards establishing the properties (S1)–(S3). Fix a feasible (and presumably optimal) solution x of the linear program (2). We show that it is possible to find an integral feasible solution X_v, $v \in V$, such that $\sum_{v \in V} X_v = O(\log d) \cdot LP$ where $LP = \sum_{v \in V} x_v$.

Let $\alpha = 300 \cdot \ln d$. We partition the vertex set as follows: Let

$$\begin{aligned} V_0 &= \{v \in V \mid x_v \leq 1/(4d)\}, \\ V_1 &= \{v \in V \mid x_v \geq 1/\alpha\}, \\ W &= V \setminus (V_0 \cup V_1), \text{ and let} \\ U &= V_0 \cup W. \end{aligned}$$

We define X_v for $v \in V$ to be independent 0-1 random variables such that

$$Pr[X_v = 1] = \begin{cases} 0 & \text{for } v \in V_0, \\ \alpha x_v & \text{for } v \in W, \\ 1 & \text{for } v \in V_1. \end{cases} \tag{3}$$

Note that X_v for $v \in V_0 \cup V_1$ are set deterministically. Thus the actual random variables are only X_v, $v \in W$.

Before defining the bad events, we need a few auxiliary observations. The next lemma, although simple, seems to be the crucial step in our proof. It shows that by removing nodes in V_1, the maximal degree decreases to a constant, and that the values x_v, $v \in W$, add up to a significant fraction in each constraint.

Lemma 3.2 *Let $u \in U$ be such that $\delta^U(u) > d$. Let $i = \delta^U(u) - d$ (i.e., $(u, N^U(u)) \in \mathcal{S}_{d+i}$). Then*

(i) $\sum_{v \in N^U(u)} x_v \geq i(1 - 1/\alpha)$.

(ii) $|N^U(u)| = d + i \leq d + d/(\alpha - 2) \leq 2d$.

(iii) $\sum_{v \in N^W(u)} x_v \geq i(1 - 1/\alpha) - 1/2 \geq i/4$.

Proof. Consider the constraint of the linear program (2) corresponding to the star $(u, N^U(u))$. Using $x_u \leq 1/\alpha$ (since $u \in U$), the constraint implies $i/\alpha + \sum_{v \in N^U(u)} x_v \geq i$, which is equivalent to (i).

Using now $x_v \leq 1/\alpha$ for all v in the sum, (i) implies $(d+i)/\alpha \geq i(1-1/\alpha)$ which is equivalent to the first inequality of (ii). The second inequality of (ii) follows by the choice of α.

Using (ii) and $x_v \leq 1/(4d)$ for $v \in V_0$, we have $\sum_{v \in N^{V_0} U(u)} x_v \leq 1/2$. Combining this with (i) and $W = U - V_0$ implies (iii). $\square$

Lemma 3.3 *Suppose that there exists $u \in U$ with $\delta^U(u) > d$. Then there exists a partitioning of the set W into classes $T_1, T_2, \ldots, T_m$ such that for each class T_j we have $|T_j| \le 3d$ and*

$$1/4 \le \sum_{v \in T_j} x_v \le 3/4 \ .$$

Proof. Using Lemma 3.2 (iii) for u, we know that $\sum_{v \in W} x_v \ge 1/4$. Furthermore, $x_v \le 1/\alpha \le 1/4$ for all $v \in W$. We can now construct the partition greedily. Add the elements $v \in W$ into the current T_i one by one. Once the sum reaches $1/4$, start a new set. If the last set has the sum smaller than $1/4$, add it to the previous one. It follows that we obtain sets T_j with $\sum_{v \in T_j} x_v$ between $1/4$ and $3/4$. Now, since for each v, $x_v \ge 1/(4d)$, it follows that $|T_j| \le (3/4)/(1/4d) = 3d$. □

We are ready to define the bad events. For the rest of the proof, fix a partitioning T_1, T_2, $\ldots$, T_m satisfying Lemma 3.3. We define two types of bad events. The first type of events help us to control the feasibility of X, the other type helps us to control the size of X (i.e., $\sum_{u \in V} X_u$).

Definition 3.4 *For a node $u \in U$ with $\delta^U(u) = d + i$, $i \ge 1$, the bad event F_u occurs if $\sum_{v \in N^W(u)} X_v < i$. The support set of F_u is $N^U(u)$.*

For a set T_j from the partitioning of W, the bad event E_j occurs if $\sum_{v \in T_j} X_v > 2\alpha \cdot \sum_{v \in T_j} x_v$. The support set of E_j is T_j.

To bound the probability of the bad events we use the following standard form of Chernoff-Hoeffding Bounds.

Lemma 3.5 (Chernoff-Hoeffding Bounds) *Consider any set of k independent binary random variables $Y_1, \ldots, Y_k$. Let $Y = \sum_{i=1}^{n} Y_i$ and $\mu = \mathbf{E}[Y]$. Then for any $\delta > 0$ it holds that*

$$\mathbf{Pr}[Y \le (1-\delta)\mu] \le \mathrm{e}^{-\mu\delta^2/2} \ ,$$

$$\mathbf{Pr}[Y \ge (1+\delta)\mu] \le \mathrm{e}^{-\mu\delta^2/3} \ .$$

Lemma 3.6 *The probability of each bad event F_u and E_j is at most $1/d^9$.*

Proof. Consider the event F_u for $u \in U$ with $\delta^U(u) = d + i$, $i \ge 1$. Let $Y = \sum_{v \in N^W(u)} X_u$. We want to prove that the probability of the event $Y < i$ is small. From the definition of X_u and Lemma 3.2 (iii) we have that $\mathbf{E}[Y] = \alpha \cdot \sum_{v \in N^W(u)} x_u \geq \alpha i/4$. Thus, by the first Chernoff-Hoeffding bound with $\delta = 1/2$,

$$\mathbf{Pr}[F_u] = \mathbf{Pr}[Y < i] \leq \mathbf{Pr}[Y < \mathbf{E}[Y]/2] \leq e^{-\alpha i/32}$$

Since $i \geq 1$ and $\alpha = 300 \cdot \ln d$, the probability is at most $1/d^9$.

Consider now the event E_j. Let $Y = \sum_{v \in T_j} X_v$. We have $\mathbf{E}[Y] = \alpha \cdot \sum_{v \in T_j} x_v \geq \alpha/4$ and E_j occurs if $Y > 2 \cdot \mathbf{E}[Y]$. Using the second Chernoff-Hoeffding bound with $\delta = 1$, we have

$$\mathbf{Pr}[E_j] = \mathbf{Pr}[Y > 2 \cdot \mathbf{E}[Y]] \leq e^{-\alpha/12},$$

which is again at most $1/d^9$. $\square$

We are almost ready to apply the Lovász Local Lemma. It remains to provide an upper bound on the dependency among the bad events.

Lemma 3.7 *Each $v \in W$ is in the support set of at most $2d+1$ bad events. Each support set of a bad event intersects at most $6d^2$ support sets of other bad events.*

Proof. Each $v \in W$ is in the support of exactly one bad event E_j. If v is in the support of F_u, which is $N^U(u)$, then $u \in N^U(v)$. By Lemma 3.2 (ii), $|N^U(v)| \leq 2d$ and the first part of the lemma follows.

Next we observe that each support set of a bad event has size at most $3d$: This follows by Lemma 3.2 (ii) for F_u and by Lemma 3.3 for E_j. Each of the $3d$ elements of the support set can be contained in at most $2d$ other support sets, and the lemma follows. $\square$

At this point we are ready to prove our main result.

Theorem 3.8 *For some $C > 0$, there exists a randomized $(C \cdot \ln d)$-approximation algorithm for* Bounded Degree Deletion.

Proof. As the first step of the algorithm, we find an optimal solution x_u of the linear program 2 using ellipsoid method. Recall that $LP = \sum_{v \in V} x_v$ is a lower bound on the optimal solution of Bounded Degree Deletion.

Next we define the distribution of random variables X_v using (3) and the bad events by Definition 3.4.

We verify the conditions of Theorem 3.1 for $D = 6d^2$. The condition (S1) on limited dependency of bad events follows from Lemma 3.7. From Lemma 3.6 it follows that the probability of each bad event is at most $1/d^9$.

Since $D^4/d^9 \geq O(1/d)$, the condition (S2) follows for d larger than a certain constant. The condition (S3) about efficient sampling of X_v's and efficient computing of the bad events is obvious from the definitions.

Now we use the algorithm guaranteed by the Theorem 3.1 to obtain an integral assignment X_v not satisfying any bad event. We output (i.e., remove from the graph) the set Z of all v such that $X_v = 1$.

We claim that Z is a feasible output of Bounded Degree Deletion. Consider any node $u \in V - Z$. We want to show that $\delta^{V-Z}(u) \leq d$. Since $V_1 \subseteq Z$, we have $V - Z \subseteq U$ and in particular $u \in U$. If $\delta^U(u) \leq d$ then we are done. Otherwise, since F_u does not occur, we know that sufficiently many neighbors $v \in W$ of u have $X_v = 1$ and are in Z, and $\delta^{V-Z}(u) \leq d$ as well.

Finally, we bound the size of Z. For each T_j, since E_j does not occur, we have $\sum_{v \in T_j} X_v \leq 2\alpha \cdot \sum_{v \in T_j} x_v$. Summing over all j we obtain $\sum_{v \in W} X_v \leq 2\alpha \cdot \sum_{v \in W} x_v$. By the definition of V_1, we have $\sum_{v \in V_1} X_v \leq \alpha \cdot \sum_{v \in V_1} x_v$. Using the fact that $X_v = 0$ for $v \in V_0$ we obtain $\sum_{v \in V} X_v \leq 2\alpha \cdot \sum_{v \in V} x_v = 2\alpha \cdot LP$. The theorem follows.

(For d which is not sufficiently large, that is, below the constant implied by the proof above, we use the $1/(d+2)$ approximation algorithm described in Section 2.) $\square$

4 The integrality gap

Theorem 4.1 *The integrality gap of the linear program (2) is $\Theta(\log d)$, for $\log d \geq 2$.*

Proof. The upper bound of $O(\log d)$ on the integrality gap is implied by Theorem 3.8 and its proof.

It remains to prove the lower bound. For every m and every $d \geq \binom{2m-1}{m}$, we describe a graph $G = (V, E)$ such that the integrality gap of the linear program (2) is $\Theta(m)$. This shows the desired bound, since for a given d, we may use $m = \lfloor \log_2 d \rfloor / 2$.

The core of G is a bipartite graph $G' = (A \cup B, E)$ where $A = \{1, \ldots, 2m\}$ and B is the set of all subsets of A of size m. There is an edge between $a \in A$ and $b \in B$ if $a \in b$. The degrees of vertices are $\delta^A(b) = m$ for each $b \in B$ and $\delta^B(a) = \binom{2m-1}{m}$ for each $a \in A$. The remaining vertices C of the graph G are leaves connected each to a single vertex $b \in B$ so that $\delta^{A \cup C}(b) = d + 1$. That means that there are $d+1-m$ leaves from C connected to each $b \in B$. There are no other vertices and edges in G. Since $d \geq \binom{2m-1}{m}$, the only vertices with degree larger than d are the vertices B.

We claim that integrality gap for G is $\Theta(m)$. More precisely, we show that the optimal solution of Bounded Degree Deletion has $m+1$ vertices while the linear program (2) has a fractional solution with value 2.

First consider the fractional solution. We put $x_a = 1/m$ for all $a \in A$ and $x_v = 0$ for all $v \in B \cup C$. The only constraints of (2) are those for $(d+1)$-stars at each $b \in B$, and those are satisfied since $\sum_{a \in N^A(b)} x_a = \delta^A(b)/m = 1$. The objective is $|A|/m = 2$.

Now consider any integral solution Z for Bounded Degree Deletion in G. We claim that $|Z| \geq m+1$. Let $W = A - Z$. If $|W| < m$ then we are done, as $|Z| \geq |A| - |W| \geq m+1$. So assume $|W| \geq m$. Consider any $b \in B$ such that $b \subseteq W$. Since Z is a feasible output it must contain b or one of its neighbors. Since $N^A(b) = b \subseteq W$ is disjoint with Z, it follows that Z contains either b or one of its neighbors in C. Thus, for each $b \subseteq W$ of size m there is a distinct element of $Z \cap (B \cup C)$ which implies that $|Z \cap (B \cup C)| \geq \binom{|W|}{m} \geq |W| + 1 - m$. Since $|Z \cap A| = 2m - |W|$, we conclude that $|Z| \geq m+1$.

For completeness, we note that any $Z \subseteq A$ with $|Z| = m+1$ is a feasible solution. Also, note that the number of vertices of the constructed G is $2^{\Theta(m)}$. □

References

[1] B. Balasundaram, S. Butenko, I. V. Hicks, and S. Sachdeva. Clique relaxations in social network analysis: The maximum k-plex problem. Manuscript, 2006.

[2] N. Baldwin, E. Chesler, S. Kirov, M. Langston, J. Snoddy, R. Williams, and B. Zhang. Computational, integrative, and comparative methods for the elucidation of genetic coexpression networks. *Journal of Biomedicine and Biotechnology*, 2:172–180, 2005.

[3] E. J. Chesler, L. Lu, S. Shou, Y. Qu, J. Gu, J. Wang, H. C. Hsu, J. D. Mountz, N. E. Baldwin, M. A. Langston, D. W. Threadgill, K. F. Manly, and R. W. Williams. Complex trait analysis of gene expression uncovers polygenic and pleiotropic networks that modulate nervous system function. *Nature Genetics*, 37:233–242, 2005.

[4] M. R. Fellows, J. Guo, H. Moser, and R. Niedermeier. A generalization of Nemhauser and Trotter's Local Optimization Theorem. In *Proc. 26th Int. Symp. on Theoretical Aspects of Computer Science (STACS)*, pages 409–420, 2009.

[5] S. G. Kolliopoulos and N. E. Young. Approximation algorithms for covering/packing integer programs. *Journal of Computer and System Sciences*, 71:495–505, 2005.

[6] C. Komusiewicz, F. Hüffner, H. Moser, and R. Niedermeier. Isolation concepts for enumerating dense subgraphs. In *Proc. 13th Ann. Int. Computing and Combinatorics Conf. (COCOON)"*, pages 140–150, 2007.

[7] N. Nishimura, P. Ragde, and D. M. Thilikos. Fast fixed-parameter tractable algorithms for nontrivial generalizations of vertex cover. *Discrete Appl. Math.*, 152(1-3):229–245, 2005.

[8] S. B. Seidman and B. L. Foster. A graph-theoretic generalization of the clique concept. *Journal of Mathematical Sociology*, 6:139–154, 1978.

[9] A. Srinivasan. Improved algorithmic versions of the Lovász Local Lemma. In *Proc. 19th Symp. on Discrete Algorithms (SODA)*, pages 611–620, 2008.