Journal of Discrete Algorithms 33 (2015) 71–80

Contents lists available at ScienceDirect

Journal of Discrete Algorithms

www.elsevier.com/locate/jda

On the complexity of making a distinguished vertex minimum or maximum degree by vertex deletion

Sounaka Mishra [a,*], Ashwin Pananjady [b,1], Safina Devi N. [a]

[a] Department of Mathematics, Indian Institute of Technology Madras, 600036, India
[b] Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, United States

ARTICLE INFO

Article history:
Received 15 January 2014
Received in revised form 24 January 2015
Accepted 15 March 2015
Available online 20 March 2015

Keywords:
Node deletion problems
Approximation algorithm
Hardness of approximation

ABSTRACT

In this paper, we investigate the approximability of two node deletion problems. Given a vertex weighted graph $G = (V, E)$ and a specified, or "distinguished" vertex $p \in V$, MDD(min) is the problem of finding a minimum weight vertex set $S \subseteq V \setminus \{p\}$ such that p becomes the minimum degree vertex in $G[V \setminus S]$; and MDD(max) is the problem of finding a minimum weight vertex set $S \subseteq V \setminus \{p\}$ such that p becomes the maximum degree vertex in $G[V \setminus S]$. These are known NP-complete problems and they have been studied from the parameterized complexity point of view in [1]. Here, we prove that for any $\epsilon > 0$, both the problems cannot be approximated within a factor $(1 - \epsilon)\log n$, unless $\mathsf{NP} \subseteq \mathsf{Dtime}(n^{\log\log n})$. We also show that for any $\epsilon > 0$, MDD(min) cannot be approximated within a factor $(1 - \epsilon)\log n$ on bipartite graphs, unless $\mathsf{NP} \subseteq \mathsf{Dtime}(n^{\log\log n})$, and that for any $\epsilon > 0$, MDD(max) cannot be approximated within a factor $(1/2 - \epsilon)\log n$ on bipartite graphs, unless $\mathsf{NP} \subseteq \mathsf{Dtime}(n^{\log\log n})$. We give an $O(\log n)$ factor approximation algorithm for MDD(max) on general graphs, provided the degree of p is $O(\log n)$. We then show that if the degree of p is $n - O(\log n)$, a similar result holds for MDD(min). We prove that MDD(max) is APX-complete on 3-regular unweighted graphs and provide an approximation algorithm with ratio 1.583 when G is a 3-regular unweighted graph. In addition, we show that MDD(min) can be solved in polynomial time when G is a regular graph of constant degree.

1. Introduction

The problems of making a distinguished vertex minimum or maximum degree by vertex deletion in undirected graphs are very natural, albeit unexplored problems in graph theory, and see a wide array of applications. We formally state these two problems as follows.

- MDD(min): Given a graph $G = (V, E)$ with a distinguished vertex $p \in V$, find a vertex set $S \subseteq V \setminus \{p\}$ of minimum size such that the vertex p is the unique vertex of minimum degree in $G[V \setminus S]$.

* Corresponding author.
E-mail addresses: sounak@iitm.ac.in (S. Mishra), ashwinpm@berkeley.edu (A. Pananjady), safina@smail.iitm.ac.in (N. Safina Devi).
[1] Work done while the author was pursuing B.Tech. at IIT Madras.

http://dx.doi.org/10.1016/j.jda.2015.03.002
1570-8667/

- MDD(max): Given a graph $G = (V, E)$ with a distinguished vertex p, find a vertex set $S \subseteq V\backslash\{p\}$ of minimum size such that the vertex $p \in V$ is the unique vertex of maximum degree in $G[V \setminus S]$.

Variants of these problems include the *weighted case*, in which we are interested in finding a vertex set S of minimum weight instead of minimum cardinality, when each vertex in G has a weight associated with it.

These problems have been previously studied in [1,2] with reference to directed graphs and electoral networks. The most natural motivation lies in competitive social networks, which are undirected, and in which the degree of a node is widely seen as a measure of its popularity, influence or importance. An agent may wish to decrease the influence of a competing agent (minimize the degree of a distinguished vertex) or increase his own influence (maximizing degree of a distinguished vertex) at minimum cost, by shielding the minimum number of other agents from the network.

Another application lies in terrorist networks studied extensively in [3,4], in which the connectivity of a particular node in the network may be decreased by targeting the minimum number of other nodes. The MDD(min) problem finds a direct application in this scenario, as well as in similar scenarios involving cartel networks.

The third major application could lie in biology – in protein networks. There have been a multitude of papers published [5,12,14] which try to correlate the parameter of a particular node in the network – such as degree, centrality, etc. – with the importance of the corresponding protein. While degree is seen as a reasonably good indicator of connectivity and influence, it may be interesting to look at how many other proteins would have to disappear from the network in order to make a particular protein influential. This is a direct application of MDD(max), and the minimum number of other proteins which need to be deleted could provide a measure of essentiality of the protein corresponding to the distinguished vertex. The research in this area has been mainly empirical so far, and this could provide another metric to judge the importance of a particular protein given its interaction network.

Both MDD(min) and MDD(max) are known to be NP-complete [1]. Previous work on these two problems involved approaches using parameterized complexity [1], but a classical complexity approach has not yet been taken as per our knowledge. In this paper, we take a classical complexity theory approach towards the problems and make the following contributions:

- We show that MDD(min) on a graph G is equivalent to MDD(max) on the graph G^c.
- We prove that both MDD(min) and MDD(max) are hard to approximate within a factor smaller than $\log n$, where n represents the number of vertices in the input graph.
- On bipartite graphs, we prove that MDD(min) and MDD(max) are hard to approximate within a factor smaller than $O(\log n)$.
- We propose an $O(\log n)$ factor approximation algorithm for MDD(max) when the input graph G satisfies a certain property. As a consequence of this, we show that if $d(p) = O(\log n)$ in the input graph G, MDD(max) is approximable within a factor of $O(\log n)$.
- We show that MDD(min) is solvable in polynomial time on k-regular graphs, as long as $k = O(1)$.
- For 3-regular unweighted graphs, we propose an approximation algorithm for MDD(max) with approximation ratio 1.583. On 3-regular bipartite graphs, we prove that MDD(max) is APX-complete.

2. Preliminaries

All the discussion in this paper concerns undirected graphs. The word *graph* is used to mean undirected graph without any ambiguity.

2.1. Notation

In a graph $G = (V, E)$, the sets $N_G(v) = \{u \in V(G) : (u, v) \in E\}$ and $N_G[v] = N_G(v) \cup \{v\}$ denote the *neighbourhood* and the *closed neighbourhood* of a vertex v in G, respectively. The *degree* of a vertex v in G is $|N_G(v)|$ (or the number of neighbours of v in graph G) and is denoted by $d_G(v)$. Note that even if $v \notin V(H)$, $d_H(v)$ could be non-zero, if $v \in V(G)$ and H is a subgraph of G. We shall use $N(v)$, $N[v]$ and $d(v)$ instead of $N_G(v)$, $N_G[v]$ and $d_G(v)$, respectively, when there is no ambiguity regarding the graph under consideration. In a similar vein, for a set of vertices S, we define $N(S) = \cup_{v\in S} N(v)$ and $N[S] = \cup_{v\in S} N[v]$.

A graph $G = (V, E)$ is called *k-regular* if $d_G(v) = k$, $\forall v \in V$. For $S \subseteq V$, $G[S]$ denotes the subgraph induced by S on G. The *complement of a graph* $G = (V, E)$ is the graph $G^c = (V, E^c)$, where $(u, v) \in E^c$ if and only if $(u, v) \notin E$, $\forall u, v \in V, u \neq v$. Unless otherwise mentioned, n denotes the number of vertices in the input graph.

In a graph $G = (V, E)$, $S \subseteq V$ is called a *dominating set* in G if $N[S] = V$. Given a graph $G = (V, E)$, an instance of MDD(max), we say that $S \subseteq V \setminus \{p\}$ is a *solution to MDD(max)* for G, if the vertex p is the maximum degree vertex in $G[V \setminus S]$. S is called a *minimal solution to MDD(max)* for G if, for each $u \in S$, $S \setminus \{u\}$ is not a solution to MDD(max) for G. A *minimum solution* to MDD(max) on graph G is a solution S to MDD(max) of minimum weight/cardinality. Similarly, a solution (and minimal solution, minimum solution) to MDD(min) for G is defined.