Information Processing Letters 116 (2016) 590–594

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

An FPT-algorithm for modifying a graph of bounded treewidth to decrease the size of its dominating set using minimum modification [☆]

Mehdy Roayaei [a], Mohammadreza Razzazi [a,b,*]

[a] Department of Computer Engineering and Information Technology, AmirKabir University of Technology, Tehran, Iran
[b] School of Computer Science, Institute for Research in Fundamental Sciences (IPM), Tehran, Iran

ARTICLE INFO

Article history:
Received 26 October 2015
Received in revised form 2 April 2016
Accepted 2 April 2016
Available online 7 April 2016
Communicated by M. Chrobak

Keywords:
Dominating set
Fixed-parameter tractability
Tree decomposition
Computational Complexity

ABSTRACT

In this paper, we study the problem of modifying a graph such that the resulting graph has a dominating set of size at most k. Solving this problem determines the minimum number of edges to be added to a given graph such that at most k vertices can dominate all vertices. We show that this problem is NP-hard, and therefore, has no polynomial-time algorithm (unless $P = NP$). Also, we show that the problem is FPT parameterized by the treewidth of the input graph. Furthermore, we show that the problem parameterized by k is W[2]-hard and belongs to W[P].

1. Introduction

One of the most well-known problems in networks, especially in computer networks, is Minimum Dominating Set: Given an undirected graph $G = (V, E)$, a dominating set is a subset S of its vertices such that for all vertices $v \in V$, either $v \in S$ or it has a neighbor in S. The problem of computing a dominating set of minimum size is called Minimum Dominating Set. The number of vertices in the smallest dominating set for G is called the domination number, $\gamma(G)$.

Although Dominating Set is W[2]-complete with respect to the parameter domination number [5], it is FPT with respect to the parameter treewidth [15]. Also, it has been shown that Dominating Set parameterized by the domination number is FPT for graphs with bounded degeneracy [14] and planar graphs [7].

This problem has diverse applications in the real world, including dominating queens, set of representatives, bus routing, computer communication networks, radio stations, social network theory, land surveying, and kernels which have been illustrated in [1] and [2]. In most of the applications, by computing a dominating set, we are actually locating our resources such as the server computers in a computer network or bus stops in a transportation network; so, minimizing the size of the dominating set is directly related to minimizing the cost of the required resources on the dominating nodes.

In this paper, we consider the situation in which there is some threshold on the number of our resources; therefore, all vertices must be dominated using at most k vertices. We call it *Domination Number Improvement*. The solution of this problem determines the minimum number of edges to be added to a given graph so that at most k vertices can dominate all vertices. Domination Number Improvement is interesting when there is some threshold

☆ This research was in part supported by a grant from School of Computer Science, Institute for Research in Fundamental Sciences (IPM) (No. CS1393-8-1).

* Corresponding author at: P.O. Box 15875-4413, #424 Hafez Avenue, AmirKabir University of Technology, Tehran, Iran. Tel.: +98 2164542732.
E-mail addresses: mroayaei@aut.ac.ir (M. Roayaei), razzazi@aut.ac.ir (M. Razzazi).

http://dx.doi.org/10.1016/j.ipl.2016.04.002
0020-0190/© 2016 Elsevier B.V. All rights reserved.

on the size of the dominating set because of a limitation on the cost or on the number of available resources [12] or when adding new edges between vertices of a graph is less expensive than adding new resources [13].

Definition 1 *(Domination Number Improvement).* Given an undirected graph $G = (V, E)$ and an integer k, find the minimum number of edges, the set $\hat{E}$, such that the graph $\hat{G} = (V, E \cup \hat{E})$ has domination number $\gamma(\hat{G}) = k$.

We show that this problem is NP-hard. Furthermore, we present an algorithm that runs in polynomial time for fixed treewidth of the input graph. More precisely, we show that Domination Number Improvement is FPT parameterized by the treewidth of the input graph. Also, we show that Domination Number Improvement parameterized by k is W[2]-hard and belongs to W[P]. W[P] is the class of all parameterized problems that can be reduced to Weighted Circuit Satisfiability by a parameterized reduction [17].

2. Main algorithm

We show that Domination Number Improvement can be reduced to k-Max Dominating Set [3] and vice versa. It is known that k-Max Dominating Set is NP-hard and it cannot be approximated with a ratio better than $1 - \frac{1}{e}$ [3]. Also, it is known that k-Max Dominating Set parameterized by the maximum number of dominated vertices is FPT [16].

Definition 2 *(k-Max Dominating Set).* Given an undirected graph $G = (V, E)$ and an integer k, find a subset $D \subseteq V$ with $|D| \leq k$, that maximizes the number of dominated vertices.

k-Max Dominating Set has applications in different networks, such as sensor networks, where bandwidth constraints limit the number of sensors we can choose to k [10], or social networks, where the goal is to find k vertices having maximum total influence in a social network [11].

Theorem 1. *An undirected graph $G = (V, E)$ and an integer k are given. There is a solution of size $n - d$ for k-Max Dominating Set, if and only if there is a solution of size d for Domination Number Improvement.*

Proof. It is clear that any instance for one problem is also an instance for the other. First, suppose that the subset $D \subseteq V$ with $|D| \leq k$ is a solution for k-Max Dominating Set that dominates $n - d$ vertices. For each non-dominated vertex, add an edge connecting it to a vertex in D. Clearly, the resulting graph has the domination number of size at most k. Therefore, the set of d new edges, namely S_d, is a solution for Domination Number Improvement.

Second, assume that the set S_d with $|S_d| = d$ is a solution for Domination Number Improvement. That is, adding the set S_d to the graph G converts it to the graph $\hat{G} = (V, E \cup S_d)$, which has a dominating set D of size at most k. Removing each edge of S_d from $\hat{G}$ can convert at most one dominated vertex to a non-dominated vertex. Therefore, after removing S_d from $\hat{G}$, the set D can dominate at least $n - d$ vertices. □

Corollary 1. *Domination Number Improvement is NP-hard.*

Corollary 2. *Any optimal solution of k-Max Dominating Set can be translated to an optimal solution of Domination Number Improvement in linear time.*

Typically, algorithms based on the tree decomposition technique consist of two major steps. The first step is responsible for finding a tree decomposition of bounded width (fixed treewidth) of the input graph, and the second step is responsible for solving the problem on that tree decomposition usually using the dynamic programming approach [4].

Definition 3 *(Tree decomposition).* Let $G = (V, E)$ be a graph. A *tree decomposition* of G is a pair $\langle \{X_i | i \in I\}, T \rangle$, where each X_i is a subset of V called a *bag*, and T is a tree with the elements of I as vertices. The following three properties must hold:

1. $\cup_{i \in I} X_i = V$;
2. for every edge $\{u, v\} \in E$, there is an $i \in I$ such that $\{u, v\} \subseteq X_i$; and
3. for all $i, j, k \in I$, if j lies on the path between i and k in T then $X_i \cap X_k \subseteq X_j$.

The width of $\langle \{X_i | i \in I\}, T \rangle$ equals $max\{|X_i| : i \in I\} - 1$. The *treewidth* of G is the minimum ω such that G has a tree decomposition of width ω.

The problem of deciding whether a graph has treewidth at most ω and then finding its corresponding tree decomposition is NP-complete for general graphs [5]. However, this problem is FPT parameterized by the treewidth of the input graph, and there is an algorithm to find the tree decompositions of *small treewidth*, which runs in linear time for fixed treewidth of the input graph [6].

From now on, assume that a tree decomposition with the treewidth ω is given. Also, k is the upper bound for the size of the dominating set.

To reduce the time complexity of our algorithm, we take advantage of a special type of tree decomposition. The tree decomposition of a graph is not unique and there may be more than one tree decomposition for a graph, each of which has different properties. A special tree decomposition that has a simple structure but attractive properties is called *nice tree decomposition* [7]:

Definition 4 *(Nice tree decomposition).* A tree decomposition $\langle \{X_i | i \in I\}, T \rangle$ is a *nice tree decomposition* if it satisfies the following properties:

1. Every bag of the tree T has at most two children.
2. If a bag i has two children j and k, then $X_i = X_j = X_k$. X_i is called a *join bag*.
3. If a bag i has one child j, then either
   (a) $|X_i| = |X_j| + 1$ and $X_j \subset X_i$, X_i is called an *insert bag*, or
   (b) $|X_i| = |X_j| - 1$ and $X_i \subset X_j$, X_i is called a *forget bag*.