Subexponential Fixed-Parameter Algorithms for Partial Vector Domination ☆,☆☆

Toshimasa Ishii[a,*], Hirotaka Ono[b], Yushi Uno[c]

[a] *Graduate School of Economics and Business Administration, Hokkaido University, Sapporo 060-0809, Japan.*
[b] *Department of Economic Engineering, Faculty of Economics, Kyushu University, Fukuoka 812-8581, Japan.*
[c] *Department of Mathematics and Information Sciences, Graduate School of Science, Osaka Prefecture University, Sakai 599-8531, Japan.*

Abstract

Given a graph $G = (V, E)$ of order n and an n-dimensional non-negative vector $\mathbf{d} = (d(1), d(2), \ldots, d(n))$, called demand vector, the vector domination (resp., total vector domination) is the problem of finding a minimum $S \subseteq V$ such that every vertex v in $V \setminus S$ (resp., in V) has at least $d(v)$ neighbors in S. The (total) vector domination is a generalization of many dominating set type problems, e.g., the (total) dominating set problem, the (total) k-dominating set problem (this k is different from the solution size), and so on, and subexponential fixed-parameter algorithms with respect to solution size for apex-minor-free graphs (so for planar graphs) are known. In this paper, we consider maximization versions of the problems; that is, for a given integer k, the goal is to find an $S \subseteq V$ with size k that maximizes the total sum of satisfied demands. For these problems, we design subexponential fixed-parameter algorithms with respect to k for apex-minor-free

☆An extended abstract of this article was presented in *Proceedings of the 3rd International Symposium on Combinatorial Optimization* (ISCO 2014), Lecture Notes in Computer Science, Vol. 8596, pp. 292–304, Springer, 2014.

☆☆This work is partially supported by KAKENHI No. 15H00853, 23500022, 24700001, 24106004, 25104521, 25106508, 26280001 and 26540005, the Kayamori Foundation of Informational Science Advancement and The Asahi Glass Foundation.

*Corresponding author

Email addresses: ishii@econ.hokudai.ac.jp (Toshimasa Ishii), hirotaka@econ.kyushu-u.ac.jp (Hirotaka Ono), uno@mi.s.osakafu-u.ac.jp (Yushi Uno)

graphs.

Keywords: (total) vector dominating set, partial dominating set, fixed-parameter tractability, branchwidth, apex-minor-free graphs.

---

1. Introduction

Given a graph $G = (V, E)$ of order n and an n-dimensional non-negative vector $\mathbf{d} = (d(1), d(2), \ldots, d(n))$, called *demand vector*, the *vector domination* (resp., *total vector domination*) is the problem of finding a minimum $S \subseteq V$ such that every vertex v in $V \setminus S$ (resp., in V) has at least $d(v)$ neighbors in S. These problems were introduced by [21], and they contain many existing problems, such as the dominating set, the total dominating set, the k-dominating set problem [16], and the total k-dominating set problem [28] (these k's are different from the solution size), and so on. Indeed, by setting $\mathbf{d} = (1, \ldots, 1)$, the (total) vector domination becomes the minimum (total) dominating set forms, and by setting $\mathbf{d} = (k, \ldots, k)$, the (total) vector dominating set becomes the (total) k-dominating set. If in the definition of total vector domination, we replace open neighborhoods with closed ones, we get the *multiple domination* [22, 23]. In this paper, we sometimes refer to these problems just as *domination problems*. Table 1 of [8] summarizes how related problems are represented in the scheme of domination problems. Many variants of the basic concepts of domination and their applications have appeared in [23, 24].

Since the vector or multiple domination (resp., total vector domination) include the setting of the ordinary dominating set problem (resp., total dominating set problem), they are obviously NP-hard, and the parameterized complexity is considered. It is well-known that the ordinary dominating set problem (resp., total dominating set problem) is $W[2]$-complete [15] (resp., as mentioned in [20]); it is unlikely to be FPT with respect to the solution size. For treewidth as another parameter, it is shown that the vector domination problem is $W[1]$-hard with respect to treewidth [2]. This result and Courcelle's meta-theorem about MSOL [10] imply that the MSOL-based algorithm is not applicable for the vector domination problem. Very recently, a polynomial-time algorithm for the vector domination of graphs with bounded clique-width has been proposed [6]. Since $cw(G) \leq 3 \cdot 2^{tw(G)-1}$ holds where $tw(G)$ and $cw(G)$ respectively denote the treewidth and clique-width of graph G [9], their polynomial-time algorithm implies the polynomial-time solvability of the vector domination problem for graphs of bounded treewidth. Also, polynomial-time algorithms for the domination problems of graphs with bounded branchwidth are independently proposed in [26, 27]. It is known that $\max\{bw(G), 2\} \leq tw(G)+1 \leq \max\{3bw(G)/2, 2\}$, where $bw(G)$ denotes the branchwidth of graph G [30]. Due to the linear relation of treewidth and branchwidth, the above results imply the polynomial-time solvability of all the domination problems (i.e., vector domination, total vector domination and multiple domination) for graphs of bounded treewidth. Furthermore, by extending the algorithms, we can show that these for apex-minor-free graphs are subexponential fixed-parameter tractable with respect to the solution size k^*, that is, there is an algorithm whose running time is $2^{O(\sqrt{k^*}\log k^*)}n^{O(1)}$. Note that the class of apex-minor-free graphs includes planar graphs.

In this paper, we consider the parameterized complexity of maximization variants of the domination problems for apex-minor-free graphs. For a given integer k, the goal is to find an $S \subseteq V$ with size k that maximizes the total sum of satisfied demands. We call the maximization problems *partial domination problems*, that is, *partial vector domination*, and so on. As the parameter, we adopt the given k itself. In the case of the ordinary dominating set or the ordinary total dominating set, there is a $2^{O(\sqrt{k^*})}n^{O(1)}$-time algorithm for planar graphs (or apex-minor-free graphs), where k^* is the size of a (total) dominating set of G. This subexponential running time is obtained by combining the following results [18]: (1) The branchwidth of planar graph G is $O(\sqrt{k^*})$ (this also holds for apex-minor-free graphs). (2) There is an algorithm whose running time is $O(3^{3bw(G)/2}|E|)$. This idea can be extended to the domination problems, and there are subexponential fixed-parameter algorithms with respect to solution size for the domination problems of apex-minor-free graphs, as mentioned above [26, 27].

In the case of partial domination problems, however, it might be difficult to bound the branchwidth of G itself by using k, because k could be much smaller than k^*. Instead, we try to choose a special S among all the optimal solutions. Roughly speaking, in this strategy, S and its neighbors are localized so that the branchwidth of the subgraph of G induced by S and its neighbors is bounded by $O(\sqrt{k})$. Then, we can expect a similar speed-up effect; the points become (i) how we localize S, and (ii) the design of a fixed-parameter algorithm whose exponent is linear of $bw(G)$. This scheme is proposed by [17], and they have succeeded to design subexponential fixed-parameter algorithms with respect to k for the partial dominating set and the partial vertex cover of apex-minor-free graphs. Their time complexities are both $2^{O(\sqrt{k})}n^{O(1)}$, and optimal in the sense of order, since even for planar graphs, there is no $2^{o(\sqrt{k})}n^{O(1)}$-time algorithm unless the Exponential Time Hypothesis (ETH) (i.e., the assumption that there is no $2^{o(n)}$-time algorithm for n-variable 3SAT [25]) fails. These properties about the lower bounds on the parameterized complexity follow since neither the ordinary (not partial) dominating set problem nor the ordinary vertex cover problem with planar graphs has a $2^{o(\sqrt{k})}n^{O(1)}$-time algorithm unless ETH fails [4]. Similarly, we can observe that by combining the reductions proposed in [19, Theorem 2] and [4, Lemma 5.1], the partial total domination problem with planar graphs also has no $2^{o(\sqrt{k})}n^{O(1)}$-time algorithm, unless ETH fails.

In this paper, we present subexponential fixed-parameter algorithms with respect to k for the partial domination problems, i.e., the partial vector domination, the partial total vector domination and the partial multiple domination, of apex-minor-free graphs. Their running times are all $2^{O(\sqrt{k}\log\min\{d^*+1,k+1\})}n^{O(1)}$, where $d^* = \max\{d(v) \mid v \in V\}$. Namely, we show that the results of [17] can be extended to vector versions. This subexponential fixed-parameter running time is obtained by a similar way as above; (i) the localization of S and (ii) the design of $2^{O(bw(G)\log\min\{d^*+1,k+1\})}n^{O(1)}$-time algorithms, whose parameters are not only $bw(G)$ but also k and d^*. It is different from the case of the partial dominating set, but it eventually yields the subexponential fixed-parameter running time. It should be noted that the running time for the partial vector domination or the partial multiple domination generalizes the result of the partial dominating set, because the dominating set is included in the case of $d^* = 1$. Also, we remark that by the above lower bounds on the parameterized complexity for the partial (total) dominating set problem, our subexponential algorithms for all of three domination problems are optimal if d^* is a constant (note that the partial dominating set problem (resp., partial total dominating set problem) is equivalent to the partial vector domination or the partial multiple domination (partial vector domination) for $\mathbf{d} = (1, 1, \ldots, 1)$).

1.1. Related Work

The dominating set problem itself is one of the most fundamental graph optimization problems, and it has been intensively and extensively studied from many points of view. In the sense that the vector or multiple domination contains the setting of not only the ordinary dominating set problem but also many variants, there are an enormous number of related studies. Here we pick some representatives up.

As a research of the domination problems from the viewpoint of the algorithm design, Cicalese, Milanič and Vaccaro gave detailed analyses of the approximability and inapproximability [7, 8]. They also provided some exact polynomial-time algorithms for special classes of graphs, such as complete graphs, trees, P_4-free graphs, and threshold graphs.

For graphs with bounded treewidth (or branchwidth), the ordinary domination problems can be solved in polynomial time. As for the fixed-parameter tractability, it is known that even the ordinary dominating set problem is W[2]-complete with respect to solution size k^*; it is unlikely to be fixed-parameter tractable [15]. In contrast, it can be solved in $O(2^{11.98\sqrt{k^*}}k^* + n^3)$ time for planar graphs, that is, it is subexponential fixed-parameter tractable [14]. The subexponent part comes from the inequality $bw(G) \leq 12\sqrt{k^*} + 9$. Behind the inequality, there is a unified property of parameters, called *bidimensionality*. Namely, the subexponential fixed-parameter algorithm of the dominating set for planar graphs (more precisely, H-minor-free graphs) is based on the bidimensionality.

Partial Dominating Set is the problem of maximizing the number of vertices to be dominated by using a given number k of vertices, and our problems are considered as its generalizations. In [1], it was shown that partial dominating set problem is FPT with respect to k for H-minor-free graphs. Later, [17] gives a subexponential FPT with respect to k for apex-minor-free graphs. Although partial dominating set is an example of problems to which the bidimensionality theory cannot be applied, they develop a technique to reduce an input graph so that its treewidth becomes $O(\sqrt{k})$.

For the (not partial) vector domination, a polynomial-time algorithm for graphs of bounded treewidth has been proposed very recently [6]. In [29], it is shown that the vector domination for ρ-degenerated graphs can be solved in $k^{O(\rho k^{*2})}n^{O(1)}$ time, if $d(v) > 0$ holds for $\forall v \in V$ (positive constraint). Since any planar graph is 5-degenerated, the vector domination for planar graphs is fixed-parameter tractable with respect to solution size, under the positive constraint. Furthermore, the case where $d(v)$ could be 0 for some v can be easily reduced to the positive case by using the transformation discussed in [2], with increasing the degeneracy by at most 1. It follows that the vector domination for planar graphs is FPT with respect to solution size k^*. For the total vector domination and multiple vector domination, [27] presents first polynomial time algorithms for graphs of bounded treewidth (or bounded branchwidth). The same paper presents first subexponential fixed-parameter algorithms with respect to k^* for apex-minor-free graphs for all the domination problems (i.e., vector, total vector, and multiple domination problems). See also [26].

Other than these, several generalized versions of the dominating set problem are also studied. (k, r)-center is the problem that asks the existence of set S of k vertices satisfying that for every vertex $v \in V$ there exists a vertex $u \in S$ such that the distance between u and v is at most r; $(k, 1)$-center corresponds to the ordinary dominating set. The (k, r)-center for planar graphs is shown to be fixed-parameter tractable with respect to k and r [11]. For $\sigma, \rho \subseteq \{0, 1, 2, \ldots\}$ and a positive integer k, $\exists[\sigma, \rho]$-dominating set is the problem that asks the existence of set S of k vertices satisfying that $|N(v) \cap S| \in \sigma$ holds for $\forall v \in S$ and $|N(v) \cap S| \in \rho$ for $\forall v \in V \setminus S$, where $N(v)$ denotes the open neighborhood of v. If $\sigma = \{0, 1, \ldots\}$ and $\rho = \{1, 2, \ldots\}$, $\exists[\sigma, \rho]$-dominating set is the ordinary dominating set problem, and if $\sigma = \{0\}$ and $\rho = \{0, 1, 2, \ldots\}$, it is the independent set. In [5], the parameterized complexity of $\exists[\sigma, \rho]$-dominating set with respect to treewidth is also considered.

The remainder of the paper is organized as follows. In Section 2, we introduce some basic notations, problem definitions and then explain the branch decomposition. Section 3 is the main part of the paper. We present fixed-parameter algorithms with respect to the branchwidth and k and show how we can localize S. Then we obtain subexponential fixed-parameter algorithms with respect to k for apex-minor-free graphs. Section 4 concludes the paper.

2. Preliminaries

A graph G is an ordered pair of its vertex set $V(G)$ and edge set $E(G)$ and is denoted by $G = (V(G), E(G))$. We assume throughout this paper that all graphs are undirected, and simple, unless otherwise stated. Therefore, an edge $e \in E(G)$ is an unordered pair of vertices u and v, and we often denote it by $e = (u, v)$. Two vertices u and v are *adjacent* if $(u, v) \in E(G)$. For a graph G, the (*open*) *neighborhood* of a vertex $v \in V(G)$ is the set $N_G(v) = \{u \in V(G) \mid (u, v) \in E(G)\}$, and the *closed neighborhood* of v is the set $N_G[v] = N_G(v) \cup \{v\}$.

For a graph $G = (V, E)$, let $\mathbf{d} = (d(v) \mid v \in V)$ be an n-dimensional non-negative vector called a *demand vector*, where $n = |V(G)|$. Then, we call a set $S \subseteq V$ of vertices a $\mathbf{d}$-*vector dominating set* (resp., $\mathbf{d}$-*total vector dominating set*) if $|N_G(v) \cap S| \geq d(v)$ holds for every vertex $v \in V \setminus S$ (resp., $v \in V$). We call a set $S \subseteq V$ of vertices a $\mathbf{d}$-*multiple dominating set* if $|N_G[v] \cap S| \geq d(v)$ holds for every vertex $v \in V$. We may drop $\mathbf{d}$ in these notations if there are no confusions.

In this paper, we consider the *partial domination problems* defined as follows.

> Partial domination problem: Given a graph $G = (V, E)$, a demand vector $\mathbf{d}$, and an integer $k \geq 0$, find a set S of vertices with cardinality at most k which maximizes the total sum $g(S)$ of demands satisfied by S.

In the setting of the *partial vector domination problem*, the function $g(S)$ is defined as

$$g(S) = \sum_{v \in S} d(v) + \sum_{v \in V \setminus S} \min\{d(v), |N_G(v) \cap S|\},$$

since each vertex $v \in S$ contributes not only 1 to the demand for each neighbor of v but also $d(v)$ to the demand for v. On the other hand, in the setting of the *partial total vector domination problem* (resp., the *partial multiple domination problem*), each vertex $v \in S$ contributes nothing (resp., only 1) to the demand for v, and hence the function $g(S)$ is defined as

$$g(S) = \sum_{v \in V} \min\{d(v), |N_G(v) \cap S|\} \text{ (resp., } \sum_{v \in V} \min\{d(v), |N_G[v] \cap S|\}).$$

Notice that for each of these domination problems, if $d(v) = 1$ for all $v \in V$, then $g(S)$ is equal to the number of vertices to be dominated by S. Hence, the partial vector domination problem and partial multiple domination problem are both generalizations of the partial dominating set problem.

A graph is called *planar* if it can be drawn in the plane without generating a crossing by two edges. A graph G has a graph H as a *minor* if a graph isomorphic to H can be obtained from G by a sequence of deleting vertices, deleting edges, or contracting edges. A graph class $\mathcal{C}$ is *minor-closed* if for each graph $G \in \mathcal{C}$, all minors of G belong to $\mathcal{C}$. A minor-closed class C is H-*minor-free* for a fixed graph H if $H \notin \mathcal{C}$. It is known that a graph is planar if and only if it has neither K_5 nor $K_{3,3}$ as a minor. An *apex graph* is a graph with a vertex v such that the removal of v leaves a planar graph. A graph class is *apex-minor-free* if it excludes some fixed apex graph. Notice that a planar graph is apex-minor-free and an apex-minor-free graph is H-minor-free.

2.1. Branch decomposition

A *branch decomposition* of a graph $G = (V, E)$ is defined as a pair $(T = (V_T, E_T), \tau)$ such that (a) T is a tree with $|E|$ leaves in which every non-leaf node has degree 3, and (b) τ is a bijection from E to the set of leaves of T. Throughout the paper, we shall use the term *node* to denote an element in V_T for distinguishing it from an element in V.

For an edge f in T, let T_f and $T \setminus T_f$ be two trees obtained from T by removing f, and E_f and $E \setminus E_f$ be two sets of edges in E such that $e \in E_f$ if and only if $\tau(e)$ is included in T_f. The *order function* $w : E(T) \to 2^V$ is defined as follows: for an edge f in T, a vertex $v \in V$ belongs to $w(f)$ if and only if there exist an edge in E_f and an edge in $E \setminus E_f$ which are both incident to v. The *width* of a branch decomposition (T, τ) is $\max\{|w(f)| \mid f \in E_T\}$, and the *branchwidth* of G, denoted by $bw(G)$, is the minimum width over all branch decompositions of G.

In general, computing the branchwidth of a given graph is NP-hard [31]. On the other hand, Bodlaender and Thilikos [3] gave a linear time algorithm which checks whether the branchwidth of a given graph is at most k or not, and if so, outputs a branch decomposition with minimum width, for any fixed k. Also, as shown in the following lemma, for H-minor-free graphs, there exists a polynomial time algorithm for computing a branch-decomposition with width $O(bw(G))$ for any fixed H.

Lemma 1. *([13]) Let G be an H-minor-free graph. Then, a branch-decomposition of G with width $O(bw(G))$ can be computed in polynomial time for any fixed H.* $\square$

This lemma follows from the constant-factor approximation algorithm of [13] for computing a tree-decomposition of minimum width in H-minor-free graphs and the property that a tree-decomposition width w can be converted to a branch-decomposition with width at most $w + 1$ [30].

Here, we introduce the following basic properties about branch decompositions, which will be utilized in the subsequent sections (see e.g., [26, 27, Lemma 2] for its proof).

Lemma 2. *Let (T, τ) be a branch decomposition of G. For a tree T, let x be a non-leaf node and $f_i = (x, x_i)$, $i = 1, 2, 3$, be an edge incident to x (note that the degree of x is three). Then, $w(f_i) \setminus (w(f_j) \cup w(f_k)) = \emptyset$ for every $\{i, j, k\} = \{1, 2, 3\}$. Hence, $w(f_i) \subseteq w(f_j) \cup w(f_k)$.*

3. Subexponential algorithm for apex-minor-free graphs

In this section, for apex-minor-free graphs, we give subexponential fixed-parameter algorithms, parameterized by k, for the partial vector domination problem, the partial total vector domination problem, and the partial multiple domination problem; namely, we show the following theorem.

Theorem 3. *Let $G = (V, E)$ be an apex-minor-free graph with $n = |V|$. Each of the partial vector domination problem, the partial total vector domination problem, and the partial multiple domination problem can be solved in $2^{O(\sqrt{k} \log \min\{d^*+1, k+1\})} n^{O(1)}$ time, where $d^* = \max\{d(v) \mid v \in V\}$.*

The proof of this theorem is separately provided in Subsections 3.1 and 3.2. In Subsection 3.1, we give a $2^{O(b \log \min\{d^*+1,k+1\})} n^{O(1)}$-time dynamic programming algorithm for each of these three problems under the assumption that a branch decomposition of a given graph G with width b is given. Note that a branch-decomposition with width $O(bw(G))$ can be computed in polynomial time by Lemma 1. Hence, in the case of $bw(G) = O(\sqrt{k})$, Theorem 3 holds. Otherwise, we show in Subsection 3.2 that if G is apex-minor-free, then we can remove in polynomial time a set I of *irrelevant* vertices from G so that at least one optimal solution is a subset of $V \setminus I$ and optimal also for the problem in $G[V \setminus I]$, and we have $bw(G[V \setminus I]) = O(\sqrt{k})$; by applying the algorithms in Subsection 3.1 to $G[V \setminus I]$, we obtain Theorem 3.

3.1. Dynamic programming algorithm for general graphs

As mentioned in the following lemma, all of the three problems can be solved in $2^{O(b \log \min\{d^*+1,k+1\})} n^{O(1)}$ time, if a branch decomposition of G with width b is given (note that the problems maximizing g_1, g_2, and g_3, defined in Lemma 4, correspond to the partial vector domination problem, the partial total vector domination problem, and the partial multiple domination problem, respectively, as observed in Section 2). These properties follow from slightly modifying the dynamic programming algorithms of [26, 27] based on a branch decomposition of a given graph for the vector domination problem, the total vector domination problem, and the multiple domination problem.

Lemma 4. *Let $G = (V, E)$ be a graph,* $\mathbf{d}$ *be a demand vector, and k be a non-negative integer. If a branch decomposition of G with width b is given, then we can compute in $2^{O(b \log \min\{d^*+1, k+1\})} n^{O(1)}$ time a set S of vertices with cardinality at most k which maximizes a function $g_i(S; G)$ on S defined as follows for each $i = 1, 2, 3$.*
(i) $g_1(S; G) = \sum_{v \in S} d(v) + \sum_{v \in V \setminus S} \min\{d(v), |N_G(v) \cap S|\}$.
(ii) $g_2(S; G) = \sum_{v \in V} \min\{d(v), |N_G(v) \cap S|\}$.
(iii) $g_3(S; G) = \sum_{v \in V} \min\{d(v), |N_G[v] \cap S|\}$.

Proof. We consider only the case of the partial vector domination problem, i.e., the case of $g_1(S; G)$ (the partial total vector domination problem and the partial multiple domination problem can be treated similarly). We here show how to modify the algorithm of [26, 27] based on a branch decomposition of a given graph for computing a minimum $\mathbf{d}$-vector dominating set in $2^{O(b \log (d^*+1))} n^{O(1)}$ time so that it can be applied to the partial vector domination.

We first sketch the algorithm of [26, 27] based on a branch decomposition (T, τ) of $G = (V, E)$. Let $w : E(T) \to 2^V$ be the corresponding order function. We regard T with a rooted tree by choosing a non-leaf node as a root. For an edge $f = (y_1, y_2) \in E(T)$ such that y_1 is the parent of y_2, let $E_f = \{e \in E \mid \tau(e) \in V(T(y_2))\}$, and G_f be the subgraph of G induced by E_f, where $T(x)$ denotes the subtree of T rooted at $x \in V(T)$. The algorithm proceeds bottom-up in T, while computing $A_f(\mathbf{c})$ satisfying the following (*) for all vectors $\mathbf{c} \in \{\top, 0, 1, 2, \ldots, d^*\}^{|w(f)|}$ on $w(f)$ for each edge f in T.

> (*) $A_f(\mathbf{c})$ is the cardinality of a minimum set $D_f(\mathbf{c}) \subseteq V(G_f)$ such that the demand of all $v \in V(G_f) \setminus w(f)$ is satisfied by $D_f(\mathbf{c})$(i.e., $v \in D_f(\mathbf{c})$ or $|N_{G_f}(v) \cap D_f(\mathbf{c})| \geq d(v)$), every vertex $v \in w(f)$ with $c(v) = \top$ belongs to $D_f(\mathbf{c})$, and every vertex $v \in w(f)$ with $c(v) = i \in \{0, 1, \ldots, d(v)\}$ satisfies $|N_{G_f}(v) \cap D_f(\mathbf{c})| \geq d(v) - i$ if $D_f(\mathbf{c})$ exists, and $A_f(\mathbf{c}) = \infty$ otherwise. (Intuitively, $D_f(\mathbf{c})$ is a minimum vector dominating set in G_f under the assumption that the demand of every vertex in $w(f)$ satisfied by $D_f(\mathbf{c})$ is restricted to $\mathbf{c}$.)

Let f be a non-leaf edge of T and f_1 and f_2 are two edges of T which are children of f, i.e., three edges f, f_1, and f_2 are incident to a common node y and f_1 and f_2 are contained in $T(y)$. By Lemma 2, we have $w(f) \subseteq w(f_1) \cup$

$w(f_2)$ and $w(f_1)\backslash w(f) = w(f_2)\backslash w(f)$, and the value $A_f(\mathbf{c})$ is computed based on $A_{f_1}(\mathbf{c_1})$ and $A_{f_2}(\mathbf{c_2})$ for a pair of vectors $\mathbf{c_1} \in \{\top, 0, 1, 2, \ldots, d^*\}^{|w(f_1)|}$ on $w(f_1)$ and $\mathbf{c_2} \in \{\top, 0, 1, 2, \ldots, d^*\}^{|w(f_2)|}$ on $w(f_2)$ such that for each vertex $v \in w(f_1) \setminus w(f) (= w(f_2) \setminus w(f))$, we have $c_1(v) = c_2(v) = \top$ (i.e., $v \in D_{f_1}(\mathbf{c_1}) \cap (w(f_1) \setminus w(f))$ if and only if $v \in D_{f_2}(\mathbf{c_2}) \cap (w(f_1) \setminus w(f))$) or $c_1(v) + c_2(v) = d(v) - |w(f_1) \cap w(f_2) \cap D_{f_1}(\mathbf{c_1}) \cap D_{f_2}(\mathbf{c_2})|$. Roughly speaking, since the possible number of vectors $\mathbf{c} \in \{\top, 0, 1, 2, \ldots, d^*\}^{|w(f)|}$ is at most $(d^* + 2)^{|w(f)|} = 2^{|w(f)| \log (d^*+2)}$ for each edge $f \in E(T)$, computing $A_f(\mathbf{c})$ for all vectors $\mathbf{c}$ on $w(f)$ takes $2^{O(b \log (d^*+1))} n^{O(1)}$ time (notice that $\log (d^* + 2) = O(\log (d^* + 1))$ by $d^* \geq 1$); the total running time turns out to be $2^{O(b \log (d^*+1))} n^{O(1)}$ by $|V(T)| = O(|E(G)|)$.

We modify this algorithm mainly in the following points (a) and (b) so that it can be applied to the partial vector domination.

(a) For each $f \in E(T)$, we compute $A'_f(\mathbf{c}, \ell)$ satisfying the following (**) for all vectors $\mathbf{c}$ on $w(f)$ such that $c(v) \in \{\top, \max\{0, d(v) - k\}, \max\{0, d(v) - k + 1\}, \ldots, d(v)\}$ for $v \in w(f)$ and all nonnegative integers $\ell \leq k$.

> (**) Let $S_f(\mathbf{c}, \ell) \subseteq V(G_f)$ be a set of vertices with cardinality ℓ which maximizes $g_1(S_f(\mathbf{c}, \ell); G_f)$, under the assumption that every vertex $v \in w(f)$ with $c(v) = \top$ belongs to $S_f(\mathbf{c}, \ell)$, and every vertex $v \in w(f)$ with $c(v) = i \in \{\max\{0, d(v) - k\}, \max\{0, d(v) - k + 1\}, \ldots, d(v)\}$ satisfies $|N_{G_f}(v) \cap S_f(\mathbf{c}, \ell)| \geq d(v) - i$ (note that due to the constraint that $|S|$ is at most k, the satisfied demand for $v \notin S_f(\mathbf{c}, \ell)$ is at most k and we need not consider the case of $c(v) \leq d(v) - k - 1$). Let $A'_f(\mathbf{c}, \ell) = g_1(S_f(\mathbf{c}, \ell); G_f)$ if $S_f(\mathbf{c}, \ell)$ exists, and $A'_f(\mathbf{c}, \ell) = -\infty$ otherwise.

(b) Let f be a non-leaf edge of T and f_1 and f_2 are two edges of T which are children of f. The value $A'_f(\mathbf{c}, \ell)$ is computed based on $A'_{f_1}(\mathbf{c_1}, \ell_1)$ and $A'_{f_2}(\mathbf{c_2}, \ell_2)$ for a pair of vectors $\mathbf{c_1}$ on $w(f_1)$ and $\mathbf{c_2}$ on $w(f_2)$ such that for each vertex $v \in w(f_1) \setminus w(f) (= w(f_2) \setminus w(f))$, we have $c_1(v) = c_2(v) = \top$ or $c_1(v) + c_2(v) \in \{d(v) - |w(f_1) \cap w(f_2) \cap S_{f_1}(\mathbf{c_1}) \cap S_{f_2}(\mathbf{c_2})|, d(v) - |w(f_1) \cap w(f_2) \cap S_{f_1}(\mathbf{c_1}) \cap S_{f_2}(\mathbf{c_2})| + 1, \ldots, 2d(v)\}$, and $\ell_1 + \ell_2 - |w(f_1) \cap w(f_2) \cap S_{f_1}(\mathbf{c_1}, \ell_1) \cap S_{f_2}(\mathbf{c_2}, \ell_2)| = \ell$ (note that in the setting of the partial domination, all demands for a vertex are not satisfied and hence we need to consider the case of $c_1(v) + c_2(v) > d(v) - |w(f_1) \cap w(f_2) \cap S_{f_1}(\mathbf{c_1}) \cap S_{f_2}(\mathbf{c_2})|$).

We analyze the time complexity of this modified algorithm. Let f be an edge in $E(T)$. The possible number of vectors $\mathbf{c}$ on $w(f)$ is at most

$(\min\{d^*, k\} + 2)^{|w(f)|}$ since $c(v) \in \{\top, \max\{0, d(v) - k\}, \max\{0, d(v) - k + 1\}, \ldots, d(v)\}$ for $v \in w(f)$; hence the term d^* in the time complexity of the original algorithm (for the vector domination) is replaced with $\min\{d^*, k\}$. For computing $A'_f(\mathbf{c}, \ell)$, we need to consider all possible cases of $(\ell_1, \ell_2, |w(f_1) \cap w(f_2) \cap S_{f_1}(\mathbf{c_1}, \ell_1) \cap S_{f_2}(\mathbf{c_2}, \ell_2)|)$ for a fixed pair $\mathbf{c_1}$ and $\mathbf{c_2}$; this part takes $O(k^3) = O(n^3)$ times the computation for the corresponding part in the original algorithm. Also, we need to consider all possible cases of $(c_1(v), c_2(v))$ with $d(v) - |w(f_1) \cap w(f_2) \cap S_{f_1}(\mathbf{c_1}) \cap S_{f_2}(\mathbf{c_2})| \leq c_1(v) + c_2(v) \leq 2d(v)$ instead of $c_1(v) + c_2(v) = d(v) - |w(f_1) \cap w(f_2) \cap S_{f_1}(\mathbf{c_1}) \cap S_{f_2}(\mathbf{c_2})|$ for $v \in (w(f_1) \setminus w(f)) \setminus S_{f_1}(\mathbf{c_1}, \ell_1)$; the number of such pairs $(c_1(v), c_2(v))$ is $O(\min\{d^*, k\}^2)$ because each of $c_1(v)$ and $c_2(v)$ takes a value between $\max\{0, d(v) - k\}$ and $d(v)$ as observed above. In the original algorithm, the number of such pairs $(c_1(v), c_2(v))$ is at most $d(v) + 1$, and hence in this part, the term d^* in the time complexity of the original algorithm is replaced with $O(\min\{d^*, k\}^2)$. Thus, we can observe that the modified algorithm can be implemented to run in $2^{O(b \log \min\{d^*+1, k+1\})} n^{O(1)}$ time. $\square$

3.2. Removal of irrelevant vertices for apex-minor-free graphs

Let G be an apex-minor-free graph. Here we show that we can remove a set I of irrelevant vertices from G so that at least one optimal solution is a subset of $V \setminus I$ and optimal also for the problem in $G[V \setminus I]$, and we have $bw(G[V \setminus I]) = O(\sqrt{k})$. In order to identify a set of irrelevant vertices, we focus on a *lexicographically smallest solution*. These ideas follow from the ones given by Fomin et al. for solving the partial vertex cover problem or partial dominating set problem [17].

Definition 5. *Given an ordering $\sigma = v_1, v_2, \ldots, v_n$ of V and two subsets X_1 and X_2 of V, we say that X_1 is* lexicographically smaller than *X_2, denoted by $X_1 \leq_\sigma X_2$, if $V_\sigma^i \cap X_1 = V_\sigma^i \cap X_2$ and $v_{i+1} \in X_1 \setminus X_2$ for some $i \in \{0, 1, \ldots, n\}$, where $V_\sigma^i = \{v_1, v_2, \ldots, v_i\}$ for $i \in \{1, 2, \ldots, n\}$ and $V_\sigma^0 = \emptyset$. For a problem P, a set $S \subseteq V$ is called a* lexicographically smallest solution *for P if for any other solution S' for P, we have $S \leq_\sigma S'$.*

Below, we show how to define an ordering σ of V and identify a set I of irrelevant vertices for the partial vector domination problem (resp., the partial total vector domination problem and the multiple domination problem) in Subsection 3.2.1 (resp., 3.2.2).

3.2.1. Vector domination

We consider the partial vector dominating set problem. Let $\sigma = v_1, v_2, \ldots, v_n$ of V be an ordering of V such that

$$d(v_1) + |N_G(v_1) \setminus V_0| \geq d(v_2) + |N_G(v_2) \setminus V_0| \geq \cdots \geq d(v_n) + |N_G(v_n) \setminus V_0|,$$

where $V_0 = \{v \in V \mid d(v) = 0\}$. Let $S_\sigma = \{v_{i_1}, v_{i_2}, \ldots, v_{i_k}\}$ be a lexicographically smallest solution for the problem where $i_1 < i_2 < \cdots < i_k$. Since each vertex $v \in S$ contributes to demands in $N_G[v]$, we have $g_1(S_\sigma; G[V_\sigma^{i_k} \cup N_G(V_\sigma^{i_k})]) = g_1(S_\sigma; G)$ (note that g_1 is defined in the statement of Lemma 4). That is, S_σ is an optimal solution also for the partial vector dominating set in $G[V_\sigma^{i_k} \cup N_G(V_\sigma^{i_k})]$; S_σ is a set of vertices with cardinality at most k maximizing g_1 also in $G[V_\sigma^{i_k} \cup N_G(V_\sigma^{i_k})]$. Thus, we have only to treat a smaller instance, instead of the original instance. Below, we consider how to find such an instance.

Now we can observe that S_σ is a $(k, 3)$-center in $G[V_\sigma^{i_k} \cup N_G(V_\sigma^{i_k})]$ (recall that a (k, r)-*center* of a graph H is a set W of vertices of H with size k such that any vertex in H is within distance r from a vertex of W).

Lemma 6. *S_σ is a $(k, 3)$-center in $G[V_\sigma^{i_k} \cup N_G(V_\sigma^{i_k})]$.*

Proof. Let $G' = G[V_\sigma^{i_k} \cup N_G(V_\sigma^{i_k})]$. For proving the lemma, we will show that $N_{G'}^2[v] \cap S_\sigma \neq \emptyset$ holds for all $v \in V_\sigma^{i_k}$, where $N_H^2[v]$ denotes the set of vertices within distance 2 from a vertex v in a graph H.

Assume for contradiction that there exists a vertex v_j with $j < i_k$ such that $N_{G'}^2[v_j] \cap S_\sigma = \emptyset$. Since G has no path with length at most 2 connecting v_j and any vertex in S_σ which goes through a vertex in $V \setminus (V_\sigma^{i_k} \cup N_G(V_\sigma^{i_k}))$, it follows that $N_G^2[v_j] \cap S_\sigma = \emptyset$ also holds. Consider the set $S' = S_\sigma \cup \{v_j\} \setminus \{v_{i_k}\}$. Note that $|S'| = |S_\sigma|$. Then, we claim that $g_1(S'; G) \geq g_1(S_\sigma; G)$ holds, i.e., S' is also optimal, which contradicts that S_σ is a lexicographically smallest solution.

This claim can be proved as follows. Observe that by deleting v_{i_k} from S_σ, the total sum $g_1(S_\sigma; G)$ of satisfied demands is decreased by at most $d(v_{i_k}) + |N_G(v_{i_k}) \setminus V_0|$. On the other hand, by $N_G^2[v_j] \cap S_\sigma = \emptyset$, an addition of v_j to $S_\sigma \setminus \{v_{i_k}\}$ increases $g_1(S_\sigma \setminus \{v_{i_k}\}; G)$ by exactly $d(v_j) + |N_G(v_j) \setminus V_0|$. It follows by the definition of σ that $g_1(S'; G) - g_1(S_\sigma; G) \geq d(v_j) + |N_G(v_j) \setminus V_0| - (d(v_{i_k}) + |N_G(v_{i_k}) \setminus V_0|) \geq 0$. $\square$

Let G be apex-minor-free. Then, the following results about algorithms for computing (k, r)-centers are known; we denote the PTAS for the problem with $r = 3$ in the following lemma by algorithm A.

Lemma 7. *([12, Corollary 5.1]) Let G be an apex-minor-free graph. Then, there is a polynomial-time approximation scheme (PTAS) for the problem of finding a minimum set W of vertices such that W is a $(|W|, r)$-center in G; for each fixed constant $\epsilon > 0$, a $(1 + \epsilon)$-approximate solution for the problem can be obtained in polynomial time.*

By utilizing algorithm A in the following manner, we can find in polynomial time a set $V_\sigma^{i'}$ of vertices for some $i' \geq i_k$ such that $G[V_\sigma^{i'} \cup N_G(V_\sigma^{i'})]$ has a $((1 + \epsilon)k, 3)$-center for a positive constant ϵ:

Step 1: Let $i := n$.
Step 2: While $G[V_\sigma^{i} \cup N_G(V_\sigma^{i})]$ does not have a $((1 + \epsilon)k$,3)-center (this can be checked by algorithm A), let $i := i - 1$.

Let i' be the value of i when this procedure halts. By $i' \geq i_k$, we have $S_\sigma \subseteq V_\sigma^{i'}$ and $g_1(S_\sigma; G[V_\sigma^{i'} \cup N_G(V_\sigma^{i'})]) = g_1(S_\sigma; G)$, and S_σ is an optimal solution also for the problem in $G[V_\sigma^{i'} \cup N_G(V_\sigma^{i'})]$.

Now, it is known that if an apex-minor-free graph has a (k, r)-center, then its branchwidth is $O(r\sqrt{k})$.

Lemma 8. *([17, Lemma 2]) Let G be an apex-minor-free graph. If G has a (k, r)-center, then the treewidth (branchwidth) of G is $O(r\sqrt{k})$.*

Since $G[V_\sigma^{i'} \cup N_G(V_\sigma^{i'})]$ has a $((1 + \epsilon)k$,3)-center, it follows that we have $bw(G[V_\sigma^{i'} \cup N_G(V_\sigma^{i'})]) = O(\sqrt{(1 + \epsilon)k}) = O(\sqrt{k})$; $V \setminus (V_\sigma^{i'} \cup N_G(V_\sigma^{i'}))$ can be regarded as a set I of irrelevant vertices mentioned above.

Thus, we can compute in polynomial time a subgraph G' of G of small branchwidth $O(\sqrt{k})$ such that an optimal solution for G is optimal also for G'.

3.2.2. Total vector domination and multiple domination

We first consider the partial total vector dominating set problem. The difference between the partial total vector domination and the partial vector domination is that when a vertex v is selected as a member in a solution, v contributes nothing to the demand of v for the former problem, but the demand $d(v)$ is satisfied for the latter problem. For this problem, by defining an ordering $\sigma = v_1, v_2, \ldots, v_n$ of V as

$$|N_G(v_1) \setminus V_0| \geq |N_G(v_2) \setminus V_0| \geq \cdots \geq |N_G(v_n) \setminus V_0|,$$

we can obtain a counterpart of Lemma 6 for the partial vector dominating set problem; namely, a lexicographically smallest solution $S_\sigma = \{v_{i_1}, v_{i_2}, \ldots, v_{i_k}\}$ for the problem is a $(k,3)$-center in $G[V_\sigma^{i_k} \cup N_G(V_\sigma^{i_k})]$, where $i_1 < i_2 < \cdots < i_k$.

Lemma 9. *S_σ is a $(k,3)$-center in $G[V_\sigma^{i_k} \cup N_G(V_\sigma^{i_k})]$.*

Proof. Let $G' = G[V_\sigma^{i_k} \cup N_G(V_\sigma^{i_k})]$. Similarly to the proof of Lemma 6, it suffices to assume that there exists a vertex v_j with $j < i_k$ such that $N_G^2[v_j] \cap S_\sigma = \emptyset$, and derive a contradiction. Now, consider the set $S' = S_\sigma \cup \{v_j\} \setminus \{v_{i_k}\}$. Note that $|S'| = |S_\sigma|$. Then, we claim that $g_2(S'; G) \geq g_2(S_\sigma; G)$ holds, i.e., S' is also optimal, which contradicts that S_σ is a lexicographically smallest solution.

This claim can be proved as follows. Observe that by deleting v_{i_k} from S_σ, the total sum $g_2(S_\sigma; G)$ of satisfied demands is decreased by at most $|N_G(v_{i_k}) \setminus V_0|$. On the other hand, by $N_G^2[v_j] \cap S_\sigma = \emptyset$, an addition of v_j to $S_\sigma \setminus \{v_{i_k}\}$ increases $g_2(S_\sigma \setminus \{v_{i_k}\}; G)$ by exactly $|N_G(v_j) \setminus V_0|$. It follows by the definition of σ that $g_2(S'; G) - g_2(S_\sigma; G) \geq |N_G(v_j) \setminus V_0| - |N_G(v_{i_k}) \setminus V_0| \geq 0$. □

The remaining parts can be treated in a similar way to the case of the partial vector dominating set. Also, we can treat the partial multiple vector dominating set, by replacing $N_G()$ with $N_G[]$ for the arguments for the partial total vector dominating set.

4. Concluding Remarks

In this paper, we have shown that for apex-minor-free graphs, the partial vector domination problem, the partial total vector domination problem, and the partial multiple domination problem are all subexponential FPT with respect to k; the time complexities are $2^{O(\sqrt{k} \log \min\{d^*+1, k+1\})} n^{O(1)}$. As mentioned in Introduction, there is no $2^{o(\sqrt{k})} n^{O(1)}$-time algorithm for the partial (total) dominating set problems even for planar graphs, unless ETH fails. Hence, when d^* is a constant, the running times for all of the above partial domination problems are best possible. It is future work to improve the gap between these upper bounds and lower bounds on the parameterized complexity for a general d^*.

Another interesting issue is to consider the problem which maximizes the number of satisfied vertices (i.e., the number of vertices v with $v \in S$ or $|N_G(v) \cap S| \geq d(v)$ in the case of the vector domination problem), instead of the total sum of satisfied demands.

Acknowledgments: We are very grateful to the anonymous referees for careful reading and suggestions.

References

[1] O. Amini, F. V. Fomin, S. Saurabh, Implicit branching and parameterized partial cover problems, Journal of Computer and System Sciences 77 (6) (2011) 1159–1171.

[2] N. Betzler, R. Bredereck, R. Niedermeier, J. Uhlmann, On bounded-degree vertex deletion parameterized by treewidth, Discrete Applied Mathematics 160 (1) (2012) 53–60.

[3] H. L. Bodlaender, D. M. Thilikos, Constructive linear time algorithms for branchwidth, in: Automata, Languages and Programming, vol. 1256 of Lecture Notes in Computer Science, Springer, 1997, pp. 627–637.

[4] L. Cai, D. Juedes, On the existence of subexponential parameterized algorithms, Journal of Computer and System Sciences 67 (4) (2003) 789–807.

[5] M. Chapelle, Parameterized complexity of generalized domination problems on bounded tree-width graphs, arXiv preprint arXiv:1004.2642.

[6] F. Cicalese, G. Cordasco, L. Gargano, M. Milanič, U. Vaccaro, Latency-bounded target set selection in social networks, Theoretical Computer Science 535 (2014) 1–15.

[7] F. Cicalese, M. Milanič, U. Vaccaro, Hardness, approximability, and exact algorithms for vector domination and total vector domination in graphs, in: Fundamentals of Computation Theory, vol. 6914 of Lecture Notes in Computer Science, Springer, 2011, pp. 288–297.

[8] F. Cicalese, M. Milanič, U. Vaccaro, On the approximability and exact algorithms for vector domination and related problems in graphs, Discrete Applied Mathematics 161 (6) (2013) 750 – 767.

[9] D. G. Corneil, U. Rotics, On the relationship between clique-width and treewidth, SIAM Journal on Computing 34 (4) (2005) 825–847.

[10] B. Courcelle, The monadic second-order logic of graphs. I. recognizable sets of finite graphs, Information and computation 85 (1) (1990) 12–75.

[11] E. D. Demaine, F. V. Fomin, M. T. Hajiaghayi, D. M. Thilikos, Fixed-parameter algorithms for (k, r)-center in planar graphs and map graphs, ACM Transactions on Algorithms (TALG) 1 (1) (2005) 33–47.

[12] E. D. Demaine, M. T. Hajiaghayi, Bidimensionality: new connections between FPT algorithms and PTASs, in: Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms, Society for Industrial and Applied Mathematics, 2005, pp. 590–601.

[13] E. D. Demaine, M. T. Hajiaghayi, K.-i. Kawarabayashi, Algorithmic graph minor theory: Decomposition, approximation, and coloring, in: Foundations of Computer Science, 2005. FOCS 2005. 46th Annual IEEE Symposium on, IEEE, 2005, pp. 637–646.

[14] F. Dorn, Dynamic programming and fast matrix multiplication, in: Algorithms–ESA 2006, vol. 4168 of Lecture Notes in Computer Science, Springer, 2006, pp. 280–291.

[15] R. G. Downey, M. R. Fellows, Fixed-parameter tractability and completeness, Cornell University, Mathematical Sciences Institute, 1992.

[16] J. F. Fink, M. S. Jacobson, n-domination in graphs, in: Graph Theory with Applications to Algorithms and Computer Science, John Wiley & Sons, Inc., New York, NY, USA, 1985, pp. 283–300.

[17] F. V. Fomin, D. Lokshtanov, V. Raman, S. Saurabh, Subexponential algorithms for partial cover problems, Information Processing Letters 111 (16) (2011) 814–818.

[18] F. V. Fomin, D. M. Thilikos, Dominating sets in planar graphs: branch-width and exponential speed-up, SIAM Journal on Computing 36 (2) (2006) 281–309.

[19] V. Garnero, I. Sau, A linear kernel for planar total dominating set, arXiv preprint arXiv:1211.0978.

[20] P. A. Golovach, J. Kratochvíl, O. Suchý, Parameterized complexity of generalized domination problems, Discrete Applied Mathematics 160 (6) (2012) 780–792.

[21] J. Harant, A. Pruchnewski, M. Voigt, On dominating sets and independent sets of graphs, Combinatorics, Probability and Computing 8 (1999) 547–553.

[22] F. Harary, T. W. Haynes, Double domination in graphs, Ars Combinatoria 55 (2000) 201–214.

[23] T. W. Haynes, S. T. Hedetniemi, P. J. Slater, Domination in graphs: advanced topics, vol. 40, Marcel Dekker, 1998.

[24] T. W. Haynes, S. T. Hedetniemi, P. J. Slater, Fundamentals of domination in graphs, Marcel Dekker, 1998.

[25] R. Impagliazzo, R. Paturi, F. Zane, Which problems have strongly exponential complexity?, Journal of Computer and System Sciences 63 (4) (2001) 512–530.

[26] T. Ishii, H. Ono, Y. Uno, (Total) vector domination for graphs with bounded branchwidth, arXiv preprint arXiv:1306.5041.

[27] T. Ishii, H. Ono, Y. Uno, (Total) vector domination for graphs with bounded branchwidth, in: LATIN 2014: Theoretical Informatics, vol. 8392 of Lecture Notes in Computer Science, Springer, 2014, pp. 238–249.

[28] A. P. Kazemi, On the total k-domination number of graphs, Discussiones Mathematicae Graph Theory 32 (3) (2012) 419–426.

[29] V. Raman, S. Saurabh, S. Srihari, Parameterized algorithms for generalized domination, in: Combinatorial Optimization and Applications, Lecture Notes in Computer Science, Springer, 2008, pp. 116–126.

[30] N. Robertson, P. D. Seymour, Graph minors. X. obstructions to tree-decomposition, Journal of Combinatorial Theory, Series B 52 (2) (1991) 153–190.

[31] P. D. Seymour, R. Thomas, Call routing and the ratcatcher, Combinatorica 14 (2) (1994) 217–241.