(Total) Vector Domination for Graphs with Bounded Branchwidth ☆,☆☆

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 $d = (d(1), d(2), \ldots, d(n))$, 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 dominating set problem, the k-tuple dominating set problem (this k is different from the solution size), and so on, and its approximability and inapproximability have been studied under this general framework. In this paper, we show that a (total) vector domination of graphs with bounded branchwidth can be solved in polynomial time. This implies that the problem is polynomially solvable also for graphs with bounded treewidth. Consequently, the (total) vector domination problem for a planar graph is subexponential fixed-parameter tractable with respect to

☆An extended abstract of this article was presented in *Proceedings of the 11th Latin American Theoretical Informatics Symposium* (LATIN 2014), Lecture Notes in Computer Science, Vol. 8392, pp. 238–249, Springer, 2014.

☆☆This work is partially supported by KAKENHI No. 23500022, 24700001, 24106004, 25104521, 25106508 and 26280001, 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)

k, where k is the size of solution.

Keywords: vector dominating set, total vector dominating set, multiple dominating set, FPT, bounded branchwidth

1. Introduction

Given a graph $G = (V, E)$ of order n and an n-dimensional non-negative vector $d = (d(1), d(2), \ldots, d(n))$, 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 generalize many existing problems, such as the minimum dominating set and the k-tuple dominating set problem (this k is different from the solution size) [22, 23], and so on. Indeed, by setting $d = (1, \ldots, 1)$, the vector domination becomes the minimum dominating set, and by setting $d = (k, \ldots, k)$, the total vector dominating set becomes the k-tuple dominating set. If in the definition of total vector domination, we replace open neighborhoods with closed ones, we get the *multiple domination*. In this paper, we sometimes refer to these problems just as *domination problems*. Table 1 of [9] summarizes how different variants of domination problems relate to one another. We also refer the interest reader to [23, 24] for different applications of domination problems.

Since the vector or multiple domination includes the setting of the ordinary dominating set problem, it is obviously NP-hard, and further it is NP-hard to approximate within $(c \log n)$-factor, where c is a positive constant, e.g., 0.2267 [1, 26]. As for the approximability, since the domination problems are special cases of a set-cover type integer programming problem, it is known that the polynomial-time greedy algorithm achieves an $O(\log n)$-approximation factor [15] and it is already optimal in terms of order. For further analyses in terms of approximability and inapproximability see [8, 9].

In this paper, we focus on another aspect of designing algorithms for domination problems, that is, the polynomial-time solvability of the domination problems for graphs of bounded treewidth or branchwidth. In [3], it is shown that the vector domination problem is $W[1]$-hard with respect to treewidth. On the other hand, from Courcelle's meta-theorem about MSOL [11], if the vector domination would be expressible in MSOL, then it would be linearly solvable for graphs with bounded treewidth, which contradicts the $W[1]$-hardness with respect to treewidth. To design a polynomial-time algorithm for the domination problems, we need to find a different way other than the typical MSOL approach.

In this paper, we present a polynomial-time algorithm for the domination problems in graphs with bounded branchwidth. The branchwidth is a measure of the "global connectivity" of a graph, and is known to be strictly related to treewidth. It is known that $\max\{bw(G), 2\} \leq tw(G) + 1 \leq \max\{3bw(G)/2, 2\}$, where $bw(G)$ and $tw(G)$ denote the branchwidth and treewidth of graph G, respectively [28]. Due to the linear relation of these two measures, polynomial-time solvability of a problem for graphs with bounded treewidth implies polynomial-time solvability of a problem for graphs with bounded branchwidth, and vice versa. Hence, our results imply that the domination problems (i.e., vector domination, total vector domination and multiple domination) can be solved in polynomial time for graphs with bounded treewidth; the polynomial-time solvability for all the problems (except the dominating set problem) in Table 1 of [9] is newly shown. Also, they answer the question by [8, 9] about the complexity status of the domination problems of graphs with bounded treewidth.

Furthermore, by using the polynomial-time algorithms for graphs of bounded treewidth, we can show that these problems for a planar graph are subexponential fixed-parameter tractable with respect to the size of the solution k, that is, there is an algorithm whose running time is $2^{O(\sqrt{k}\log k)}n^{O(1)}$. To our best knowledge, these are the first fixed-parameter algorithms for the total vector domination and multiple domination, whereas the vector domination for planar graphs has been shown to be FPT [27]. For the latter case, our algorithm greatly improves the running time.

Note that the polynomial-time solvability of the vector domination problem for graphs of bounded treewidth has been independently shown very recently [7]. They considered a further generalization of the vector domination problem, and gave a polynomial-time algorithm for graphs of bounded clique-width. Since $cw(G) \leq 3 \cdot 2^{tw(G)-1}$ holds where $cw(G)$ denotes the clique-width of graph G ([10]), their polynomial-time algorithm implies the polynomial-time solvability of the vector domination problem for graphs of bounded treewidth and bounded branchwidth.

1.1. Related Work

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, and hence it is unlikely to be fixed-parameter tractable [17]. 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 [16]. The subexponential part comes from the inequality $bw(G) \le 12\sqrt{k} + 9$, where k is the size of a dominating set of G. Behind the inequality, there is a unified property of parameters, called *bidimensionality* [14]. Namely, the subexponential fixed-parameter algorithm of the dominating set for planar graphs (more precisely, H-minor-free graphs [13]) is based on the bidimensionality.

A maximization version of the ordinary dominating set is also considered. *Partial Dominating Set* is the problem of maximizing the number of vertices to be dominated by using a given number k of vertices. In [2], it was shown that partial dominating set problem is FPT with respect to k for H-minor-free graphs. Later, [18] gives a subexponential FPT with respect to k for apex-minor-free graphs, also a superclass of planar graphs. Although partial dominating set is an example of problems to which the bidimensionality theory cannot be applied, the authors of [18] develop a technique to reduce an input graph so that its treewidth becomes $O(\sqrt{k})$.

For the vector domination, a polynomial-time algorithm for graphs of bounded treewidth has been proposed very recently [7], as mentioned before. In [27], 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 [3], while increasing the degeneracy by at most 1. It follows that the vector domination for planar graphs is FPT with respect to solution size k. However, for the total vector domination and multiple domination, neither polynomial time algorithm for graphs of bounded treewidth nor fixed-parameter algorithm for planar graphs has been known.

Other than these, several generalized versions of the dominating set problem are also studied. (k, r)-center problem is the one 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 [12]. For $\sigma, \rho \subseteq \{0, 1, 2, \dots\}$ 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 $\sigma = \{0, 1, \dots\}$ and $\rho = \{1, 2, \dots\}$, $\exists[\sigma, \rho]$-dominating set is the ordinary dominating set problem, and if $\sigma = \{0\}$ and $\rho = \{0, 1, 2, \dots\}$, it is the independent set. In [6], the parameterized complexity of $\exists[\sigma, \rho]$-dominating set with respect to treewidth is also considered.

1.2. Our Results

Our results are summarized as follows:

- We present a polynomial-time algorithm for the vector domination of graph $G = (V, E)$ with bounded branchwidth. The running time is roughly $O(n^{6bw(G)+2})$.
- We present polynomial-time algorithms for the total vector domination and multiple domination of graph G with bounded branchwidth. The running time is roughly $O(2^{9bw(G)/2}\ n^{6bw(G)+2})$.
- Let G be a planar graph. Then, we can check in $O(n^3 + \min\{k + 2, d^* + 2\}^{40\sqrt{k}+34} n)$ time whether G has a vector dominating set with cardinality at most k or not, where $d^* = \max\{d(v) \mid v \in V\}$.
- Let G be a planar graph. Then, we can check in $O(n^3 + 2^{30\sqrt{k}+51/2} \min\{k + 2, d^* + 2\}^{40\sqrt{k}+34} n)$ time whether G has a total vector dominating set and a multiple dominating set with cardinality at most k or not.

It should be noted that it is actually possible to design directly polynomial time algorithms for graphs with bounded treewidth, but they are slower than the ones for graphs with bounded branchwidth; from this reason, we design branch decomposition-based algorithms.

As far as the authors know, the second and fourth results give the first polynomial time algorithms and the first fixed-parameter algorithm for the total vector domination and multiple domination of graphs with bounded branchwidth (or treewidth) and planar graphs, respectively. As for the vector domination, we give an $O(n^{6bw(G)+2})$-time algorithm, whose running time is $O(n^{6(tw(G)+1)+2})$ in terms of the treewidth, whereas the recent paper [7] gives an $O(cw(G)|\sigma|(n + 1)^{5cw(G)})$-time algorithm, where $|\sigma|$ is the encoding length of k-expression used in the algorithm, and is bounded by a polynomial in the input size for fixed k. Since $cw(G) \leq 3 \cdot 2^{tw(G)-1}$ holds, this is an $O(2^{tw(G)}|\sigma|(n+1)^{7.5 \cdot 2^{tw(G)}})$-time algorithm.

Also, the third result shows that the vector domination of planar graphs is subexponential FPT with respect to k, and it greatly improves the running time of existing $k^{O(k^2)}n^{O(1)}$-time algorithm ([27]). It was shown in [5] that for the ordinary dominating set problem (equivalently, the vector domination (or multiple domination) with $d = (1, 1, \ldots, 1)$) in planar graphs, there is no $2^{o(\sqrt{k})}n^{O(1)}$-time algorithm unless the Exponential Time Hypothesis (i.e., the assumption that there is no $2^{o(n)}$-time algorithm for n-variable 3SAT [25]) fails. Hence, in this sense, our algorithm in third result (or the fourth results for the multiple domination) is optimal if d^* is a constant.

The third and fourth results give subexponential fixed-parameter algorithms of the domination problems for planar graphs. It should be noted that the domination problems themselves do not have the bidimensionality, mentioned in the previous subsection, due to the existence of the vertices with demand 0. Instead, by reducing irrelevant vertices, we obtain a similar inequality about the branchwidth and the solution size of the domination problems, which leads to the subexponential fixed-parameter algorithms.

The remainder of the paper is organized as follows. In Section 2, we introduce some basic notations and then explain the branch decomposition. Section 3 is the main part of the paper, and presents our dynamic programming based algorithms for the considered problems. Section 4 explains how we extend the algorithms of Section 3 to fixed-parameter algorithms for planar graphs.

2. Preliminaries

A graph G is an ordered pair of its vertex set $V(G)$ and edge set $E(G)$ and is denoted by $G = (V(G), E(G))$. Let $n = |V(G)|$ and $m = |E(G)|$. 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 $d = (d(v) \mid v \in V)$ be an n-dimensional non-negative vector. Then, we call a set $S \subseteq V$ of vertices a *d-vector dominating* *set* (resp., *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 *d-multiple dominating set* if $|N_G[v] \cap S| \geq d(v)$ holds for every vertex $v \in V$. We may drop d in these notations if there are no confusions.

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 [30]. On the other hand, Bodlaender and Thilikos [4] gave a linear time algorithm for any fixed k which checks whether the branchwidth of a given graph is at most k or not, and if so, outputs a branch decomposition of width at most k. Also, as shown in the following lemma, it is known that for planar graphs, it can be done in polynomial time for any given k, where a graph is called *planar* if it can be drawn in the plane without generating a crossing by two edges.

Lemma 1. *Let G be a planar graph.*
(i) (*[30]*) *It can be checked in $O(n^2)$ time whether $bw(G) \leq k$ or not for a given integer k.*
(ii) (*[20]*) *A branch decomposition of G with width $bw(G)$ can be constructed in $O(n^3)$ time.* □

Here, we introduce the following basic properties about branch decompositions, which will be used in the subsequent sections.

Lemma 2. *Let (T, τ) be a branch decomposition of G.*

(i) For 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, $w(f_i) \subseteq w(f_j) \cup w(f_k)$.
(ii) Let f be an edge of T, V_1 be the set of all end-vertices of edges in E_f, and V_2 be the set of all end-vertices of edges in $E \setminus E_f$. Then, $(V_1 \setminus w(f)) \cap (V_2 \setminus w(f)) = \emptyset$ holds. Also, there is no edge in E connecting a vertex in $V_1 \setminus w(f)$ and a vertex in $V_2 \setminus w(f)$.

Proof. (i) Without loss of generality, assume that $E_{f_1} \cap E_{f_2} = \emptyset$, $E_{f_2} \cap E_{f_3} = \emptyset$, $E_{f_3} \cap E_{f_1} = \emptyset$, and $E_{f_1} \cup E_{f_2} \cup E_{f_3} = E$. Let $v \in w(f_1)$ be a vertex. From the definition of $w(f_1)$, there exist two edges $e \in E_{f_1}$ and $e' \in E \setminus E_{f_1}$ such that both of e and e' are incident to v. If $e' \in E_{f_2}$ (resp., $e' \in E_{f_3}$), then $v \in w(f_2)$ (resp., $v \in w(f_3)$) also holds. Thus, we can observe that there is no vertex in $w(f_i) \setminus (w(f_j) \cup w(f_k))$ for every $\{i, j, k\} = \{1, 2, 3\}$.

(ii) By definition of $w(f)$, we have $V_1 \cap V_2 = w(f)$, and hence $(V_1 \setminus w(f)) \cap (V_2 \setminus w(f)) = \emptyset$.

Assume by contradiction that there exists an edge $e = (u_1, u_2) \in E$ such that $u_1 \in V_1 \setminus w(f)$ and $u_2 \in V_2 \setminus w(f)$. If we assume that $e \in E_f$ without loss of generality, then $u_2 \in V_1 \setminus w(f)$ also holds, which contradicts $(V_1 \setminus w(f)) \cap (V_2 \setminus w(f)) = \emptyset$. $\square$

3. Domination problems in graphs of bounded branchwidth

In this section, we propose dynamic programming algorithms for the vector domination problem, the total vector domination problem, and the multiple domination problem, by utilizing a branch decomposition of a given graph. The techniques are based on the one developed by Fomin and Thilikos for solving the dominating set problem with bounded branchwidth [19]. Throughout this section, for a given graph $G = (V, E)$, the demand of each vertex $v \in V$ is denoted by $d(v)$, and let $d^* = \max\{d(v) \mid v \in V\}$.

3.1. Vector domination

In this subsection, we consider the vector domination problem, and show the following theorem.

Theorem 3. *If a branch decomposition of G with width b is given, a minimum vector dominating set in G can be found in $O((d^* + 2)^b \{(d^* + 1)^2 + 1\}^{b/2} m)$ time.*

The above theorem needs the assumption that a branch decomposition of G with width b is given. Hence, for solving the vector domination problem, we need to consider how we obtain a branch decomposition of G. There exists an $O(2^{b \lg 27} n^2)$-time algorithm that given a graph G and an integer b, reports $bw(G) \geq b$, or outputs a branch decomposition of G with width at most $3b$ [29, 13]. Thus, the time to find a branch decomposition with width at most $3bw(G)$ is $O(\log bw(G) 2^{bw(G) \lg 27} n^2)$, and we have the following corollary.

Corollary 4. *A minimum vector dominating set in G can be found in $O((d^*+2)^{3bw(G)}\{(d^*+1)^2+1\}^{3bw(G)/2} n^2)$ time.* $\square$

For proving Theorem 3, we will give a dynamic programming algorithm that finds a minimum vector dominating set in G in $O((d^*+2)^b\{(d^*+1)^2+1\}^{b/2} m)$ time, based on a branch decomposition of G.

Let (T', τ) be a branch decomposition of $G = (V, E)$ with width b, and $w' : E(T') \to 2^V$ be the corresponding order function. Let T be the tree from T' by inserting two nodes r_1 and r_2, deleting one arbitrarily chosen edge $(x_1, x_2) \in E(T')$, adding three new edges (r_1, r_2), (x_1, r_2), and (x_2, r_2); namely, $T = (V(T') \cup \{r_1, r_2\}, E(T') \cup \{(r_1, r_2), (x_1, r_2), (x_2, r_2)\} \setminus \{(x_1, x_2)\})$. Here, we regard T as a rooted tree with root r_1. Let $w(f) = w'(f)$ for every $f \in E(T) \cap E(T')$, $w(x_1, r_2) = w(x_2, r_2) = w'(x_1, x_2)$, and $w(r_1, r_2) = \emptyset$.

Let $f = (y_1, y_2) \in E(T)$ be an edge in T such that y_1 is the parent of y_2. Let $T(y_2)$ be the subtree of T rooted at y_2, $E_f = \{e \in E \mid \tau(e) \in V(T(y_2))\}$, and G_f be the subgraph of G induced by E_f. Note that $w(f) \subseteq V(G_f)$ holds, since each vertex in $w(f)$ is an end-vertex of some edge in E_f by definition of the order function w'. In the following, each vertex $v \in w(f)$ will be assigned one of the following $d(v)+2$ colors $\{\top, 0, 1, 2, \ldots, d(v)\}$. The meaning of the color of a vertex v is as follows: for a vertex set (possibly, a vector dominating set) D,

- $\top$ means that $v \in D$.
- $i \in \{0, 1, \ldots, d(v)\}$ means that $v \notin D$ and $|N_{G_f}(v) \cap D| \geq d(v) - i$.

Notice that a vertex colored by $i > 0$ may need to be dominated by some vertices in $V \setminus V(G_f)$ for the feasibility. Given a coloring $c \in \{\top, 0, 1, 2, \ldots, d^*\}^{|w(f)|}$, let $D_f(c) \subseteq V(G_f)$ be a vertex set with the minimum cardinality satisfying the following (1)–(3), where $c(v)$ denotes the color assigned to a vertex $v \in V$

on coloring c on $w(f)$:

$$c(v) = \top \text{ if and only if } v \in D_f(c) \cap w(f). \quad (1)$$

$$\text{If } c(v) = i, \text{ then } v \in w(f) \setminus D_f(c)$$
$$\text{and } |N_{G_f}(v) \cap D_f(c)| \geq d(v) - i. \quad (2)$$

$$|N_{G_f}(v) \cap D_f(c)| \geq d(v) \text{ holds}$$
$$\text{for every vertex } v \in V(G_f) \setminus (w(f) \cup D_f(c)). \quad (3)$$

Intuitively, $D_f(c)$ is a minimum vector dominating set in G_f under the assumption that the color for every vertex in $w(f)$ is restricted to c. Note that a vertex in $w(f)$ is allowed not to meet its demand in G_f, because it can be dominated by some vertices in $V \setminus V(G_f)$. Also note that every vertex in $V(G_f) \setminus w(f)$ is not adjacent to any vertex in $V \setminus V(G_f)$ by Lemma 2(ii), and it needs to be dominated by vertices only in $V(G_f)$ for the feasibility. We define $A_f(c)$ as $A_f(c) = |D_f(c)|$ if $D_f(c)$ exists and $A_f(c) = \infty$ otherwise.

Our dynamic programming algorithm proceeds bottom-up in T, while computing $A_f(c)$ for all $c \in \{\top, 0, 1, 2, \ldots, d^*\}^{|w(f)|}$ for each edge f in T. We remark that $w(r_1, r_2) = \emptyset$ and $G_{(r_1,r_2)} = G$ for the root edge (r_1, r_2), and hence there is no coloring for $w(r_1, r_2)$ (we denote the coloring for $w(r_1, r_2)$ by $\emptyset$). Since $D_f(c)$ dominates all vertices in $V(G_f) \setminus w(f)$ for each edge f in T and each coloring $c \in \{\top, 0, 1, 2, \ldots, d^*\}^{|w(f)|}$, it follows that $D_{(r_1,r_2)}(\emptyset)$ is a vector dominating set in G. Furthermore, from the minimality of $A_{(r_1,r_2)}(\emptyset)$, $D_{(r_1,r_2)}(\emptyset)$ is a minimum vector dominating set in G and $A_{(r_1,r_2)}(\emptyset)$ is its cardinality. The algorithm consists of two types of procedures: one is for leaf edges and the other is for non-leaf edges, where a *leaf edge* denotes an edge incident to a leaf of T.

Procedure for leaf edges: In the first step of the algorithm, we compute $A_f(c)$ for each edge f incident to a leaf of T. Then, for all colorings $c \in \{\top, 0, 1, 2, \ldots, d^*\}^{|w(f)|}$, let $A_f(c)$ be the number of vertices colored by $\top$ if $D_f(c)$ exists and G_f and c satisfy (1) – (3), and $A_f(c) = \infty$ otherwise.

Let f be a leaf edge incident to a leaf node x in T and $e = (v_1, v_2)$ be the edge in G with $\tau(e) = x$. Then, notice that we have $w(f) = \{v_i\}$ if the degree of v_j is 1 for $\{i, j\} = \{1, 2\}$, and $w(f) = \{v_1, v_2\}$ otherwise, and that $V(G_f) = \{v_1, v_2\}$. Hence, for a fixed c, we can check in $O(1)$ time if (1) – (3) hold. This step takes $O((d^* + 2)^2)$ time per leaf edge.

Procedure for non-leaf edges: After the above initialization step, we visit non-leaf edges of T from leaves to the root of T. Let $f = (y_1, y_2)$ be a non-leaf edge of T such that y_1 is the parent of y_2, y_3 and y_4 are the children of y_2, and $f_1 = (y_2, y_3)$ and $f_2 = (y_2, y_4)$. Now we have already obtained $A_{f_j}(c')$ for all $c' \in \{\top, 0, 1, 2, \ldots, d^*\}^{|w(f_j)|}$, $j = 1, 2$. By Lemma 2(i), we have $w(f) \subseteq w(f_1) \cup w(f_2)$, $w(f_1) \subseteq w(f_2) \cup w(f)$, and $w(f_2) \subseteq w(f) \cup w(f_1)$; let $X_1 = w(f) \setminus w(f_2)$, $X_2 = w(f) \setminus w(f_1)$, $X_3 = w(f) \cap w(f_1) \cap w(f_2)$, and $X_4 = w(f_1) \setminus w(f)$ $(= w(f_2) \setminus w(f))$.

We say that a coloring $c \in \{\top, 0, 1, 2, \ldots, d^*\}^{|w(f)|}$ of $w(f)$ is *formed* from a coloring c_1 of $w(f_1)$ and a coloring c_2 of $w(f_2)$ if the following (P1)–(P5) hold.

(P1) For every $v \in X_1 \cup X_2 \cup X_3$ with $c(v) = \top$,

> (a) For every $v \in X_1 \cup X_3$, $c_1(v) = \top$ if and only if $c(v) = \top$.
> (b) For every $v \in X_2 \cup X_3$, $c_2(v) = \top$ if and only if $c(v) = \top$.

(P2) For every $v \in X_4$, $c_1(v) = \top$ if and only if $c_2(v) = \top$.

(P3) For every $v \in X_j \setminus D_{c_1,c_2}$ where $j \in \{1, 2\}$ and $D_{c_1,c_2} = \{v \in X_1 \cup X_2 \cup X_3 \cup X_4 \mid c_1(v) = \top \text{ or } c_2(v) = \top\}$,

> If $c(v) = i$, then $c_j(v) = \min\{d(v), i + |D_{c_1,c_2} \cap N_{G_f}(v) \cap X_{j'}|\}$ where $j' \in \{1, 2\} \setminus \{j\}$.
> (Intuitively, if $v \in X_j \setminus D_{c_1,c_2}$ needs to be dominated by at least $d(v) - i$ vertices in G_f, then at least $\max\{0, d(v) - i - |D_{c_1,c_2} \cap N_{G_f}(v) \cap X_{j'}|\}$ vertices from $V(G_{f_j})$ are necessary.)

(P4) For every $v \in X_3 \setminus D_{c_1,c_2}$,

> If $c(v) = i$, then $c_1(v) = \min\{d(v), i + |D_{c_1,c_2} \cap N_{G_f}(v) \cap X_2| + i_1\}$ and $c_2(v) = \min\{d(v), i + |D_{c_1,c_2} \cap N_{G_f}(v) \cap X_1| + i_2\}$ for some non-negative integers i_1, i_2 with $i_1 + i_2 = \max\{0, d(v) - i - |D_{c_1,c_2} \cap N_{G_f}(v)|\}$.
> (Intuitively, if $v \in X_3 \setminus D_{c_1,c_2}$ needs to be dominated by at least $d(v) - i$ vertices in G_f, then at least $\max\{0, d(v) - i - |D_{c_1,c_2} \cap N_{G_f}(v)|\}$ vertices from $(V(G_{f_1}) \setminus w(f_1)) \cup (V(G_{f_2}) \setminus w(f_2))$ are necessary for dominating v. If i_1 (resp., i_2) vertices among those vertices belong to $V(G_{f_2}) \setminus w(f_2)$ (resp., $V(G_{f_1}) \setminus w(f_1)$), then at least $\max\{0, d(v) - i - |D_{c_1,c_2} \cap N_{G_f}(v) \cap X_{j'}| - i_j\}$ vertices from $V(G_{f_j})$ are necessary for $\{j, j'\} = \{1, 2\}$.)

(P5) For every $v \in X_4 \setminus D_{c_1,c_2}$,

$c_1(v) = \min\{d(v), |D_{c_1,c_2} \cap N_{G_f}(v) \cap X_2| + i_1\}$ and $c_2(v) = \min\{d(v), |D_{c_1,c_2} \cap N_{G_f}(v) \cap X_1| + i_2\}$ for some non-negative integers i_1, i_2 with $i_1 + i_2 = \max\{0, d(v) - |D_{c_1,c_2} \cap N_{G_f}(v)|\}$. (This case can be treated in a similar way to (P4).)

The following two lemmas show that there exist a coloring c_1 of $w(f_1)$ and a coloring c_2 of $w(f_2)$ forming c such that $D_{f_1}(c_1) \cup D_{f_2}(c_2)$ satisfies (1)–(3) and $|D_{f_1}(c_1) \cup D_{f_2}(c_2)| = A_f(c)$. Namely, we have $A_f(c) = \min\{A_{f_1}(c_1) + A_{f_2}(c_2) - |D_{c_1,c_2} \cap (X_3 \cup X_4)| \mid c_1, c_2 \text{ forms } c\}$.

Lemma 5. *Let $c \in \{\top, 0, 1, 2, \ldots, d^*\}^{|w(f)|}$ be a coloring of $w(f)$. If a coloring c_1 of $w(f_1)$ and a coloring c_2 of $w(f_2)$ forms c, then $D_{f_1}(c_1) \cup D_{f_2}(c_2)$ satisfies* (1)–(3) *for f.*

Proof. We denote $D_{f_1}(c_1) \cup D_{f_2}(c_2)$ by D', and $D' \cap (X_1 \cup X_2 \cup X_3 \cup X_4)$ by D'_{c_1,c_2}. Clearly, (1) holds, since $v \in D' \cap w(f)$ if and only if $c(v) = \top$ by (P1).

We next show that D' satisfies (2). Let v be a vertex in $X_1 \setminus D' = X_1 \setminus D'_{c_1,c_2}$. From the above (P3), we have $|N_{G_{f_1}}(v) \cap D'| \geq d(v) - i - |D'_{c_1,c_2} \cap N_{G_f}(v) \cap X_2|$. It follows that $|N_{G_f}(v) \cap D'| \geq |N_{G_{f_1}}(v) \cap D'| + |D'_{c_1,c_2} \cap N_{G_f}(v) \cap X_2| \geq d(v) - i$. Also, the case of $v \in X_2 \setminus D'$ can be treated similarly.

Let v be a vertex in $X_3 \setminus D' = X_3 \setminus D'_{c_1,c_2}$. Since $|N_{G_f}(v) \cap D'| \geq |N_{G_f}(v) \cap D'_{c_1,c_2}|$ clearly holds, we have only to consider the case of $|N_{G_f}(v) \cap D'_{c_1,c_2}| < d(v) - i$. From (P4), we have $|N_{G_{f_1}}(v) \cap D'| \geq \max\{0, d(v) - i - |D'_{c_1,c_2} \cap N_{G_f}(v) \cap X_2| - i_1\}$ and $|N_{G_{f_2}}(v) \cap D'| \geq \max\{0, d(v) - i - |D'_{c_1,c_2} \cap N_{G_f}(v) \cap X_1| - i_2\}$ where $i_1 + i_2 = d(v) - i - |D'_{c_1,c_2} \cap N_{G_f}(v)|$ (note that $i_1 + i_2 > 0$ from the assumption of this case). Notice that $(V(G_{f_1}) \setminus w(f_1)) \cap (V(G_{f_2}) \setminus w(f_2)) = \emptyset$ by Lemma 2(ii). It follows that $|N_{G_f}(v) \cap D'| \geq |N_{G_{f_1}}(v) \cap D'| + |N_{G_{f_2}}(v) \cap D'|$ $-|N_{G_f}(v) \cap D'_{c_1,c_2} \cap (X_3 \cup X_4)| \geq 2(d(v) - i) - |N_{G_f}(v) \cap D'_{c_1,c_2}| - i_1 - i_2$ $= d(v) - i$.

We finally show that D' satisfies (3). Let v be a vertex in $X_4 \setminus D'$. Since $|N_{G_f}(v) \cap D'| \geq |N_{G_f}(v) \cap D'_{c_1,c_2}|$ clearly holds, we have only to consider the case of $|N_{G_f}(v) \cap D'_{c_1,c_2}| < d(v)$. From (P5), we have $|N_{G_{f_1}}(v) \cap D'| \geq \max\{0, d(v) - |D'_{c_1,c_2} \cap N_{G_f}(v) \cap X_2| - i_1\}$ and $|N_{G_{f_2}}(v) \cap D'| \geq \max\{0, d(v) - |D'_{c_1,c_2} \cap N_{G_f}(v) \cap X_1| - i_2\}$ where $i_1 + i_2 = d(v) - |D'_{c_1,c_2} \cap N_{G_f}(v)| > 0$. Hence, we have $|N_{G_f}(v) \cap D'| \geq |N_{G_{f_1}}(v) \cap D'| + |N_{G_{f_2}}(v) \cap D'|$ $-|N_{G_f}(v) \cap D'_{c_1,c_2} \cap (X_3 \cup X_4)| = 2d(v) - |N_{G_f}(v) \cap D'_{c_1,c_2}| - i_1 - i_2 = d(v)$. Also, it follows from the definition of $D_{f_j}(c_j)$ that $v \in V(G_{f_j}) \setminus w(f_j)$ satisfies (3) for $j = 1, 2$. $\square$

Lemma 6. *Let $c \in \{\top, 0, 1, 2, \ldots, d^*\}^{|w(f)|}$ be a coloring of $w(f)$ such that $D_f(c)$ exists. Then, there exist a coloring c_1 of $w(f_1)$ and a coloring c_2 of $w(f_2)$ forming c such that $|D_{f_1}(c_1) \cup D_{f_2}(c_2)| \leq A_f(c)$.*

Proof. For each vertex $v \in w(f_j)$, $j = 1, 2$, let

$$c_j(v) = \begin{cases} \top & \text{if } v \in D_f(c), \\ \min\{d(v), c(v) + |(N_{G_f}(v) \cap D_f(c)) \setminus V(G_{f_j})|\} & \\ & \text{if } v \in X_j \setminus D_f(c), \\ \max\{0, d(v) - |N_{G_{f_j}}(v) \cap D_f(c)|\} & \\ & \text{if } v \in X_3 \cup X_4 \setminus D_f(c). \end{cases}$$

For $v \in X_j \setminus D_f(c)$, we have $|N_{G_f}(v) \cap D_f(c)| = |N_{G_{f_j}}(v) \cap D_f(c)| + |(N_{G_f}(v) \cap D_f(c)) \setminus V(G_{f_j})| \geq d(v) - c(v)$, since $D_f(c)$ satisfies (2). Hence, $|N_{G_{f_j}}(v) \cap D_f(c)| \geq \max\{0, d(v) - c(v) - |(N_{G_f}(v) \cap D_f(c)) \setminus V(G_{f_j})|\} = d(v) - c_j(v)$ for all $v \in w(f_j) \setminus D_f(c)$. It follows from that the minimality of $A_{f_j}(c_j)$ implies that $|D_f(c) \cap V(G_{f_j})| \geq A_{f_j}(c_j)$; hence, $A_f(c) \geq |D_{f_1}(c_1) \cup D_{f_2}(c_2)|$. On the other hand, c_1 and c_2 does not necessarily form c. Below, we show that there exist a coloring c'_1 of $w(f_1)$ and a coloring c'_2 of $w(f_2)$ forming c such that $c'_j(v) \geq c_j(v)$ for every $v \in w(f_j) \setminus D_f(c)$ for $j = 1, 2$. Note that $D_{f_j}(c_j)$ satisfies (1)–(3) also for c'_j, since $|N_{G_{f_j}}(v) \cap D_{f_j}(c)| \geq d(v) - c_j(v) \geq d(v) - c'_j(v)$ for every $v \in w(f_j) \setminus D_f(c)$. Hence, from the minimality of $|D_{f_j}(c'_j)|$, we have $A_f(c) \geq |D_{f_1}(c_1) \cup D_{f_2}(c_2)| \geq |D_{f_1}(c'_1) \cup D_{f_2}(c'_2)|$, which proves this lemma.

We can construct such c'_1, c'_2 as follows. First let $c'_j(v) = c_j(v)$ for all $v \in X_1 \cup X_2 \cup D_f(c)$; c'_1 and c'_2 satisfy (P1) and (P2) in the definition of a coloring c formed by c_1 and c_2. By Lemma 2(ii), every $v \in X_j$ satisfies $(N_{G_f}(v) \cap D_f(c)) \setminus V(G_{f_j}) = N_{G_f}(v) \cap D_f(c) \cap X_{j'}$ for $\{j, j'\} = \{1, 2\}$. Hence, $c'_j(v) (= c_j(v))$ for $v \in X_j \setminus D_f(c)$, $j = 1, 2$ satisfies (P3).

Let $v \in X_3 \setminus D_f(c)$. Since $D_f(c)$ satisfies (2), we have $|N_{G_f}(v) \cap D_f(c)| \geq d(v) - c(v)$. Now from construction of c_1 and c_2, the value i'_1 (resp., i'_2) corresponding to i_1 (resp., i_2) in (P4) in the definition of c formed by c_1 and c_2 is $\max\{0, d(v) - |N_{G_{f_1}}(v) \cap D_f(c)| - c(v) - |N_{G_f}(v) \cap X_2 \cap D_f(c)|\}$ (resp., $\max\{0, d(v) - |N_{G_{f_2}}(v) \cap D_f(c)| - c(v) - |N_{G_f}(v) \cap X_1 \cap D_f(c)|\}$). It follows that $i'_1 + i'_2 \leq \max\{0, d(v) - c(v) - |N_{G_f}(v) \cap D_f(c) \cap (X_1 \cup X_2 \cup X_3 \cup X_4)|\}$ (note that the final inequality follows from $|N_{G_f}(v) \cap D_f(c)| \geq d(v) - c(v)$).

Let $v \in X_4 \setminus D_f(c)$. Since $D_f(c)$ satisfies (2), we have $|N_{G_f}(v) \cap D_f(c)| \geq d(v)$. From construction of c_1 and c_2, the value i'_1 (resp., i'_2) corresponding to i_1 (resp., i_2) in (P5) in the definition of c formed by c_1 and c_2 is $\max\{0, d(v) - |N_{G_{f_1}}(v) \cap D_f(c)| - |N_{G_f}(v) \cap X_2 \cap D_f(c)|\}$ (resp., $\max\{d(v) - |N_{G_{f_2}}(v) \cap D_f(c)| - |N_{G_f}(v) \cap X_1 \cap D_f(c)|\}$). It follows that $i'_1 + i'_2 \leq \max\{0, d(v) - |N_{G_f}(v) \cap D_f(c) \cap (X_1 \cup X_2 \cup X_3 \cup X_4)|\}$.

Consequently, we can construct a coloring c'_1 of $w(f_1)$ and a coloring c'_2 of $w(f_2)$ forming c such that $c'_j(v) \geq c_j(v)$ for every $v \in X_3 \cup X_4 \setminus D_f(c)$ and $c'_j(v) = c_j(v)$ for every $v \in D_f(c) \cup X_1 \cup X_2$ for $j = 1, 2$ by increasing i'_1 or i'_2 for each vertex $v \in X_3 \cup X_4 \setminus D_f(c)$ so that $i'_1 + i'_2$ becomes equal to $\max\{0, d(v) - c(v) - |N_{G_f}(v) \cap D_f(c) \cap (X_1 \cup X_2 \cup X_3 \cup X_4)|\}$ (resp., $\max\{0, d(v) - |N_{G_f}(v) \cap D_f(c) \cap (X_1 \cup X_2 \cup X_3 \cup X_4)|\}$) if $v \in X_3$ (resp., $v \in X_4$). $\square$

Thus, for all colorings $c \in \{\top, 0, 1, 2, \ldots, d^*\}^{|w(f)|}$, we can compute $A_f(c)$ from the information of f_1 and f_2. By repeating these procedure bottom-up in T, we can find a minimum vector dominating set in G.

Here, for a fixed c, we analyze the time complexity for computing $A_f(c)$. Let $D_c = \{v \in w(f) \mid c(v) = \top\}$, $x_j = |X_j|$ for $j = 1, 2, 3, 4$, $z_3 = |X_3 \setminus D_c|$. Under the assumption that X_4 is colored by a fixed coloring c_4, the number of pairs of a coloring c_1 of $w(f_1)$ and a coloring c_2 of $w(f_2)$ forming c is at most $(d^* + 1)^{z_3}(d^* + 1)^{z_4}$ where z_4 denotes the number of vertices in X_4 not colored by $\top$ in c_4, since the number of pairs (i_1, i_2) in (P4) or (P5) is at most $d^* + 1$ for each vertex in $X_3 \setminus D_c$ or each vertex in X_4 not colored by $\top$.

Hence, for an edge f, the number of pairs forming c is at most $(d^* + 2)^{x_1+x_2} \sum_{z_3=0}^{x_3} \binom{x_3}{z_3} (d^* + 1)^{z_3} \sum_{z_4=0}^{x_4} \binom{x_4}{z_4} (d^* + 1)^{z_4} (d^* + 1)^{z_3} (d^* + 1)^{z_4} = (d^* + 2)^{x_1+x_2} \{(d^* + 1)^2 + 1\}^{x_3+x_4}$ in total. Now we have $x_1 + x_2 + x_3 \leq b$, $x_1 + x_3 + x_4 \leq b$, and $x_2 + x_3 + x_4 \leq b$ (recall that b is the width of (T', τ)). By considering a linear programming problem which maximizes $(x_1+x_2) \log(d^* + 2) + (x_3 + x_4) \log\{(d^* + 1)^2 + 1\}$ subject to these inequalities, we can observe that $(d^* + 2)^{x_1+x_2}\{(d^* + 1)^2 + 1\}^{x_3+x_4}$ attains the maximum when $x_1 = x_2 = x_4 = b/2$ and $x_3 = 0$. Thus, it takes in total $O((d^* + 2)^b\{(d^* + 1)^2 + 1\}^{b/2})$ time to compute $A_f(c)$ for all colorings c of $w(f)$.

Since $|E(T)| = O(m)$ and the initialization step takes $O((d^* + 2)^2 m)$ time in total, we can obtain $A_{(r_1,r_2)}(c)$ in $O(((d^* + 2)^b\{(d^* + 1)^2 + 1\}^{b/2} m)$ time.

Summarizing the arguments given so far, we have shown Theorem 3.

3.2. Total vector domination and multiple domination

We consider the total vector domination problem. The difference between the total vector domination and the vector domination is that each vertex selected as a member in a dominating set needs to be dominated or not. Hence, we will modify the following parts (I)–(III) in the algorithm for vector domination given in the previous subsection so that each vertex selected as a member in a dominating set also satisfies its demand.

(I) Color assignments: Let $f \in E(T)$ be an edge in a branch decomposition T of G. We will assign to each vertex $v \in w(f)$ an ordered pair (ℓ, i) of colors, $\ell \in \{\top, \bot\}$, $i \in \{0, 1, \dots, d(v)\}$, where $\top$ means that v belongs to the dominating set, $\bot$ means that v does not belong to the dominating set, and and i means that v is dominated by at least $d(v) - i$ vertices in G_f.

(II) Conditions for $D_f(c)$: For a coloring $c \in (\{\top, \bot\} \times \{0, 1, 2, \dots, d^*\})^{|w(f)|}$, we modify (1)–(3) as follows, where let $c(v) = (c^1(v), c^2(v))$:

$c^1(v) = \top$ if and only if $v \in D_f(c) \cap w(f)$.
If $c^2(v) = i$, then $|N_{G_f}(v) \cap D_f(c)| \geq d(v) - i$.
$|N_{G_f}(v) \cap D_f(c)| \geq d(v)$ holds for every vertex $v \in V(G_f) \setminus w(f)$.

(III) Definition of a coloring c formed by c_1 and c_2: For a coloring $c \in (\{\top, \bot\} \times \{0, 1, 2, \dots, d^*\})^{|w(f)|}$, we modify (P1)–(P5) as follows:
(P1') For every $v \in X_1 \cup X_2 \cup X_3$ with $c^1(v) = \top$ (resp., $c^1(v) = \bot$),

(a) If $v \in X_1 \cup X_3$, then $c_1^1(v) = \top$ (resp., $c_1^1(v) = \bot$).
(b) If $v \in X_2 \cup X_3$, then $c_2^1(v) = \top$ (resp., $c_2^1(v) = \bot$).

(P2') For every $v \in X_4$, $c_1^1(v) = \top$ (resp., $c_1^1(v) = \bot$) if and only if $c_2^1(v) = \top$ (resp., $c_2^1(v) = \bot$).
(P3') For every $v \in X_j$ where $\{j, j'\} = \{1, 2\}$ and $D_{c_1,c_2} = \{v \in X_1 \cup X_2 \cup X_3 \cup X_4 \mid c_1^1(v) = \top \text{ or } c_2^1(v) = \top\}$,

If $c^2(v) = i$, then $c_j^2(v) = \min\{d(v), i + |D_{c_1,c_2} \cap N_{G_f}(v) \cap X_{j'}|\}$.

(P4') For every $v \in X_3$,

If $c^2(v) = i$, then $c_1^2(v) = \min\{d(v), i + |D_{c_1,c_2} \cap N_{G_f}(v) \cap X_2| + i_1\}$ and $c_2^2(v) = \min\{d(v), i + |D_{c_1,c_2} \cap N_{G_f}(v) \cap X_1| + i_2\}$ for some non-negative integers i_1, i_2 with $i_1 + i_2 = \max\{0, d(v) - i - |D_{c_1,c_2} \cap N_{G_f}(v) \cap (X_1 \cup X_2 \cup X_3 \cup X_4)|\}$.

(P5') For every $v \in X_4$,

$c_1^2(v) = \min\{d(v), |D_{c_1,c_2} \cap N_{G_f}(v) \cap X_2| + i_1\}$ and $c_2^2(v) = \min\{d(v), |D_{c_1,c_2} \cap N_{G_f}(v) \cap X_1| + i_2\}$ for some non-negative integers i_1, i_2 with $i_1 + i_2 = \max\{0, d(v) - |D_{c_1,c_2} \cap N_{G_f}(v) \cap (X_1 \cup X_2 \cup X_3 \cup X_4)|\}$.

We analyze the time complexity of this modified algorithm. Similarly to the case of the vector domination, the total running time is dominated by total complexity for computing $A_f(c)$ for non-leaf edges f.

Let f be a non-leaf edge of T and x_i, $i = 1, 2, 3, 4$ and z_4 be defined as Subsection 3.1. The number of pairs of a coloring c_1 of $w(f_1)$ and a coloring c_2 of $w(f_2)$ forming c is at most $(d^* + 1)^{x_3} \sum_{z_4=0}^{x_4} \binom{x_4}{z_4} (d^* + 1)^{x_4} (d^* + 1)^{x_4}$ since the number of pairs (i_1, i_2) in (P4') or (P5') is at most $d^* + 1$ for each vertex in $X_3 \cup X_4$. Hence, for an edge f, the number of pairs forming c is at most $\{2(d^*+1)\}^{x_1+x_2} \sum_{z_3=0}^{x_3} \binom{x_3}{z_3} (d^*+1)^{x_3} (d^*+1)^{x_3} \sum_{z_4=0}^{x_4} \binom{x_4}{z_4} (d^*+1)^{x_4} (d^*+1)^{x_4} = \{2(d^*+1)\}^{x_1+x_2} \{2(d^*+1)^2\}^{x_3+x_4}$ in total. Since $x_1 + x_2 + x_3 \le b$, $x_1 + x_3 + x_4 \le b$, and $x_2 + x_3 + x_4 \le b$, it follows that $(x_1 + x_2) \log(2d^* + 2) + (x_3 + x_4) \log\{2(d^*+1)^2\}$ attains the maximum when $x_1 = x_2 = x_4 = b/2$ and $x_3 = 0$. Thus, it takes in $O(2^{3b/2}(d^* + 1)^{2b})$ time to compute $A_f(c)$ for all colorings c of $w(f)$. Namely, we obtain the following theorem.

Theorem 7. *If a branch decomposition of G with width b is given, a minimum total vector dominating set in G can be found in $O(2^{3b/2}(d^* + 1)^{2b} m)$ time.* □

Also, by replacing $N_G()$ with $N_G[]$ in the modification for total vector domination, we can obtain the following theorem for the multiple domination problems.

Theorem 8. *If a branch decomposition of G with width b is given, a minimum multiple dominating set in G can be found in $O(2^{3b/2}(d^* + 1)^{2b} m)$ time.* □

4. Subexponential fixed parameter algorithm for planar graphs

We consider the problem of checking whether a given graph G has a d-vector dominating set with cardinality at most k. As mentioned in Subsection 1.1, if G is ρ-degenerated, then the problem can be solved in $k^{O(\rho k^2)} n^{O(1)}$ time. Since a planar graph is 5-degenerated, it follows that the problem with a planar graph can be solved in $k^{O(k^2)}n^{O(1)}$ time. In this section, we give a subexponential fixed-parameter algorithm, parameterized by k, for a planar graph; namely, we will show the following theorem.

Theorem 9. *If G is a planar graph, then we can check in* $O(\min\{2^{b^* \lg 27}n^2 + (\min\{d^*, k\} + 2)^{3(b^*+1)}\{(\min\{d^*, k\} + 1)^2 + 1\}^{3(b^*+1)/2}n,\ n^3 + (\min\{d^*, k\} + 2)^{b^*}\{(\min\{k, d^*\} + 1)^2 + 1\}^{b^*/2}n\})$ *time whether G has a d-vector dominating set with cardinality at most k or not, where* $b^* = \min\{12\sqrt{k+z} + 9, 20\sqrt{k} + 17\}$ *and* $z = |\{v \in V \mid d(v) = 0\}|$.

This time complexity is roughly $O(\min\{2^{O(\sqrt{k})}n^2, n^3\} + 2^{O(\sqrt{k}\log k)}n)$, which is subexponential with respect to k; this improves the running time of the previous fixed-parameter algorithm.

Let $V_0 = \{v \in V \mid d(v) = 0\}$ and $z = |V_0|$. In [19, Lemma 2.2], it was shown that if a planar graph G' has an ordinary dominating set (i.e., a (1,1,. . . ,1)-vector dominating set) with cardinality at most k, then $bw(G') \leq 12\sqrt{k} + 9$. This bound is based on the *bidimensionality* [14], and was used to design the subexponential fixed-parameter algorithm with respect to k for the ordinary dominating set problem. In the case of our domination problems, however, it is difficult to say that they have the bidimensionality, due to the existence of V_0 vertices. Instead, we give a similar bound on the branchwidth not w.r.t k but w.r.t $k + z$ as follows: For any (total, multiple) d-vector dominating set D of G ($|D| \leq k$), $D \cup V_0$ is an ordinary dominating set of G, and this yields $bw(G) \leq 12\sqrt{k+z} + 9$.

Actually, it is also possible to exclude z from the parameters, though the coefficient of the exponent becomes larger. To this end, we use the notion of $(k, 2)$-center. Recall that a (k, r)-*center* of G' is a set W of vertices of G' with size k such that any vertex in G' is within distance r from a vertex of W. For a (k, r)-center, a similar bound on the branchwidth is known: if a planar graph G' has a (k, r)-center, then $bw(G') \leq 4(2r+1)\sqrt{k} + 8r + 1$ ([12, Theorem 3.2]). Here, we use this bound. We can assume that for $v \in V_0$, $N_G(v) \not\subseteq V_0$ holds, because $v \in V_0$ satisfying $N_G(v) \subseteq V_0$ is never selected as a member of any optimal solution; it is *irrelevant*, and we can remove it. That is, every vertex in V_0 has at least one neighbor from $V \setminus V_0$. Then, for any (total, multiple) d-vector dominating set D of G ($|D| \leq k$), D is a $(k, 2)$-center of G. This is because any vertex in $V \setminus V_0$ is adjacent to a vertex in D and any vertex in V_0 is adjacent to a vertex in $V \setminus V_0$. Thus, we have $bw(G) \leq 20\sqrt{k} + 17$.

In summary, we have the following lemma.

Lemma 10. *Assume that G is a planar graph without irrelevant vertices, i.e., $N_G(v) \nsubseteq V_0$ holds for each $v \in V_0$. Then, if G has a (total, multiple) vector dominating set with cardinality at most k, then we have $bw(G) \leq \min\{12\sqrt{k+z}+9, 20\sqrt{k}+17\}$.* □

Combining this lemma with the algorithm in Subsection 3.1, we can check whether a given graph has a vector dominating set with cardinality at most k according to the following steps 1 and 2:

Step 1: Let $b^* = \min\{12\sqrt{k+z}+9, 20\sqrt{k}+17\}$. Check whether the branch-width of G is at most b^*. If so, then construct a branch decomposition with width at most $3(b^*+1)$ and go to Step 2, and otherwise halt after outputting 'NO'.

Step 2: Apply the dynamic programming algorithm in Subsection 3.1 to find a minimum vector dominating set for G.

As mentioned after Theorem 3, Step 1 can be executed in $O(2^{b^* \lg 27} n^2)$ time. Hence, by Theorem 3 and the fact that any planar graph G' satisfies $|E(G')| = O(|V(G')|)$, it follows that the running time of this procedure is $O(2^{b^* \lg 27} n^2 + (d^*+2)^{3(b^*+1)}\{(d^*+1)^2+1\}^{3(b^*+1)/2} n)$. Also, note that by Lemma 1, for a planar graph, we can check in $O(n^2)$ time whether $bw(G) \leq b^*$ or not, and a branch decomposition with width at most b^* can be constructed in $O(n^3)$ time. By applying these properties to Step 1, we can observe that this procedure can be implemented to run in $O(n^3 + (d^*+2)^{b^*}\{(d^*+1)^2 + 1\}^{b^*/2} n)$ time. Hence, in the case of $d^* \leq k$, Theorem 9 has been proved.

The case of $d^* > k$ can be reduced to the case of $d^* \leq k$ by the following standard kernelization method, which proves Theorem 9. Assume that $d^* > k$. Let $V_{\max}(d)$ be the set of vertices v with $d(v) = d^*$. For the feasibility, we need to select each vertex $v \in V_{\max}(d)$ as a member in a vector dominating set. Hence, if $|V_{\max}(d)| > k$, then it turns out that G has no vector dominating set with cardinality at most k. Assume that $|V_{\max}(d)| \leq k$. Then, it is not difficult to see that we can reduce an instance $I(G, d, k)$ with G, d, and k to an instance $I(G', d', k')$ such that $G' = G \setminus V_{\max}(d)$ (i.e., G' is the graph obtained from G by deleting $V_{\max}(d)$), $d'(v) = \max\{0, d(v) - |N_G(v) \cap V_{\max}(d)|\}$ for all vertices $v \in V(G')$, and $k' = \max\{0, k - |V_{\max}(d)|\}$. Based on this observation, we can reduce $I(G, d, k)$ to an instance $I(G'', d'', k'')$ with $\max\{d''(v) \mid v \in V(G'')\} \leq k'' \leq k$ or output 'YES' or 'NO' in the following manner:

(a) After setting $G' := G$, $d' := d$, and $k' := k$, repeat the procedures (b1)–(b3) while $k' < d'^*(= \max\{d'(v) \mid v \in V(G')\})$.

(b1) If $k' < |V_{\max}(d')|$, then halt after outputting 'NO.'
(b2) If $k' \geq |V_{\max}(d')|$ and $V(G') = V_{\max}(d')$, then halt after outputting 'YES.'
(b3) Otherwise after setting $G'' := G' \setminus V_{\max}(d')$, $d''(v) := \max\{0, d'(v) - |N_{G'}(v) \cap V_{\max}(d')|\}$ for each $v \in V(G'')$, and $k'' := \max\{0, k' - |V_{\max}(d')|\}$, redefine G'', d'', and k'' as G', d', and k', respectively.

We remark that this kernelization can be applied to a general graph (not restricted to a planar graph) and its running time is $O(n + m)$.

Next, we consider the total vector domination problem and the multiple domination problem. For these problems, since all vertices $v \in V$ need to be dominated by $d(v)$ vertices, the condition that $d^* \leq k$ is necessary for the feasibility. Similarly, we have the following theorem by Theorems 7 and 8.

Theorem 11. *Assume that a given graph G is planar, and let $b^* = \min\{12\sqrt{k+z} + 9, 20\sqrt{k} + 17\}$.*
(i) We can check in $O(\min\{2^{b^* \lg 27} n^2 + 2^{9(b^*+1)/2} (\min\{d^*, k\} + 2)^{6(b^*+1)} n, n^3 + 2^{3b^*/2} (\min\{d^*, k\} + 2)^{2b^*} n\})$ time whether G has a total vector dominating set with cardinality at most k or not.
(ii) We can check in $O(\min\{2^{b^* \lg 27} n^2 + 2^{9(b^*+1)/2} (\min\{d^*, k\} + 2)^{6(b^*+1)} n, n^3 + 2^{3b^*/2} (\min\{d^*, k\} + 2)^{2b^*} n\})$ time whether G has a multiple dominating set with cardinality at most k or not. □

Before concluding this section, we mention that the above results can be extended to *apex-minor-free graphs*, a superclass of planar graphs. An *apex graph* is a graph with a vertex v such that the removal of v leaves a planar graph. 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 is apex-minor-free if it does not contain any graph which has some fixed apex graph as a minor. For apex-minor-free graphs, the following lemma is known.

Lemma 12. *([18, Lemma 2]) Let G be an apex-minor-free graph. If G has a (k, r)-center, then the treewidth of G is $O(r\sqrt{k})$.*

From this lemma, the linear relation of treewidth and branchwidth, and the $2^{O(bw(G))}n^2$ -time algorithm for computing a branch decomposition with width $O(bw(G))$, we obtain the following corollary.

Corollary 13. *We can check in* $2^{O(\sqrt{k}\log k)}n^2$ *time whether an apex-minor-free graph* G *has a (total, multiple) vector dominating set with cardinality at most* k *or not.*

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

References

[1] N. Alon, D. Moshkovitz, S. Safra, Algorithmic construction of sets for k-restrictions, ACM Transactions on Algorithms (TALG) 2 (2) (2006) 153–177.

[2] 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.

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

[4] 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.

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

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

[7] F. Cicalese, G. Cordasco, L. Gargano, M. Milanič, U. Vaccaro, Latency-bounded target set selection in social networks, in: The Nature of Computation. Logic, Algorithms, Applications, Springer, 2013, pp. 65–77.

[8] 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 Berlin Heidelberg, 2011, pp. 288–297.

[9] 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.

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

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

[12] E. D. Demaine, F. V. Fomin, M. 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.

[13] E. D. Demaine, F. V. Fomin, M. Hajiaghayi, D. M. Thilikos, Subexponential parameterized algorithms on bounded-genus graphs and H-minor-free graphs, Journal of the ACM (JACM) 52 (6) (2005) 866–893.

[14] E. D. Demaine, M. Hajiaghayi, The bidimensionality theory and its algorithmic applications, The Computer Journal 51 (3) (2008) 292–302.

[15] G. Dobson, Worst-case analysis of greedy heuristics for integer programming with nonnegative data, Mathematics of Operations Research 7 (4) (1982) 515–531.

[16] F. Dorn, Dynamic programming and fast matrix multiplication, in: Algorithms–ESA 2006, Springer, 2006, pp. 280–291.

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

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

[19] 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.

[20] Q.-P. Gu, H. Tamaki, Optimal branch-decomposition of planar graphs in $O(n^3)$ time, ACM Transactions on Algorithms (TALG) 4 (3) (2008) 30.

[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 New York, 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] C. Lund, M. Yannakakis, On the hardness of approximating minimization problems, Journal of the ACM (JACM) 41 (5) (1994) 960–981.

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

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

[29] N. Robertson, P. D. Seymour, Graph minors. XIII. the disjoint paths problem, Journal of Combinatorial Theory, Series B 63 (1) (1995) 65–110.

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