Hardness, approximability, and exact algorithms for vector domination and total vector domination in graphs

Ferdinando Cicalese[1], Martin Milanič[2⋆], and Ugo Vaccaro[1]

[1]Dipartimento di Informatica ed Applicazioni, University of Salerno, 84084 Fisciano (SA), Italy
{cicalese,uv}@dia.unisa.it
[2]FAMNIT and PINT, University of Primorska, 6000 Koper, Slovenia
martin.milanic@upr.si

Abstract. We consider two graph optimization problems called vector domination and total vector domination. In vector domination one seeks a small subset S of vertices of a graph such that any vertex outside S has a prescribed number of neighbors in S. In total domination, the requirement is extended to all vertices of the graph. We prove that these problems cannot be approximated to within a factor of $c \log n$, for suitable constant c, unless every problem in NP is solvable in slightly super-polynomial time. We also show that two natural greedy strategies have approximation factor $O(\log \Delta(G))$, where $\Delta(G)$ is the maximum degree of the graph G. We also provide exact polynomial time algorithms for several classes of graphs. Our results extend and unify several results previously known in the literature.

1 Introduction

The concept of domination in graphs has been extensively studied, both in structural and algorithmic graph theory, because of its numerous applications to a variety of areas. Informally, a vertex of a graph is said to dominate itself and all of its neighbors. Generally, one seeks small sets that dominate the whole graph. Domination naturally arises in facility location problems, in problems involving finding sets of representatives, in monitoring communication or electrical networks, and in land surveying. The two books [9] [10] discuss the main results and applications of domination in graphs. Many variants of the basic concepts of domination have appeared in the literature. Again, we refer to [9] [10] for a survey of the area.

In this paper we provide hardness results and approximation algorithms for an interesting variant of the basic concept of domination, firstly introduced in

⋆ Supported in part by "Agencija za raziskovalno dejavnost Republike Slovenije", research program P1-0285.

[8]. Here, a subset of vertices S is said to dominate a vertex v if either $v \in S$, or there are in S a *prescribed* number of neighbors of x (see below for formal definitions). Again, one seeks small subsets that dominate (in this new sense) the whole vertex set of the graph.

Main Definitions. For a graph $G = (V, E)$ and a vertex $v \in V$, denote by $N(v)$ the set of neighbors of v, by $d(v)$ the degree of v, and by $\Delta(G)$ the maximum degree of any vertex in G. A *dominating set* in a graph $G = (V, E)$ is a subset S of the graph's vertex set such that every vertex not in the set has a neighbor in the set. A *total dominating set* in G is a subset $S \subseteq V$ such that every vertex of the graph has a neighbor in the set: for every $v \in V$ there exists a vertex $u \in S$ such that $uv \in E$.

The *vector domination* is the following problem: Given a graph $G = (V, E)$, and a vector $(k_v : v \in V)$ such that for all $v \in V$, $k_v \in \{0, 1, \ldots, d(v)\}$, find a *vector dominating set* S of minimum size, that is, a set $S \subseteq V$ minimizing $|S|$ and such that $|S \cap N(v)| \geq k_v$ for all $v \in V \backslash S$. The *total vector domination* is the problem of finding a minimum-sized *total vector dominating set*, that is, a set $S \subseteq V$ such that $|S \cap N(v)| \geq k_v$ for all $v \in V$.

Of special interest for us will be also the following special case of vector domination: For $0 \leq q < 1$, a *q-dominating set* in G is a subset $S \subseteq V$ such that every vertex not in the set has more than a q-fraction of its neighbors in the set: for every $v \in V \setminus S$, it holds that $|N(v) \cap S| > q|N(v)|$. For $0 \leq q < 1$, a *total q-dominating set* in G is is a subset $S \subseteq V$ such that every vertex has more than a q-fraction of its neighbors in the set: for every $v \in V$, it holds that $|N(v) \cap S| > q|N(v)|$. By $\gamma(G)$ ($\gamma^q(G)$, $\gamma_t(G)$, $\gamma_t^q(G)$) we denote the minimum size of a dominating (q-dominating, total dominating, total q-dominating) set in G. The problem of finding (for a fixed $0 \leq q < 1$) in a given graph a dominating (q-dominating, total q-dominating) set of minimum size will be referred to simply as the *domination* (*q-domination*, *total q-domination*). Notice that for graphs G, such that $\Delta(G)$ satisfies $q < 1/\Delta(G)$, the (total) q-dominating sets of G correspond to the graph's (total) dominating sets. At the other extreme, for graphs such that $q \geq 1 - 1/\Delta(G)$, the q-dominating sets of G correspond to the graph's vertex covers.

Clearly, the (total) q-domination corresponds to the special case of the (total) vector domination, in which $k_v = \lfloor q \cdot d(v) \rfloor + 1$ for all $v \in V$. In fact, we shall mainly use q-domination for our inapproximability results, and provide algorithmic results in terms of the more general problem of vector domination.

Our Results and Related Work. We first provide two natural greedy algorithms for vector domination and total vector domination in general graphs, having approximation factor of $H_{2\Delta(G)}$ and $H_{\Delta(G)}$, respectively. Subsequently, we prove that above results are essentially best possible, in the sense that both the q-domination and its total variant are inapproximable within an $O(\log|V(G)|)$ factor, unless NP $\subseteq$ DTIME($n^{O(\log\log n)}$). Notice that our inapproximability result is provided for *any* fixed $0 \leq q < 1$, hence it is not subsumed by the standard domination and total domination problems (except for $q = 0$).

We individuate special classes of graphs for which vector domination and total vector domination can be optimally solved in polynomial time. More specifically, we provide polynomial time algorithms for computing minimum size vector domination sets for and total vector domination sets for complete graphs, trees, P_4-free graphs and threshold graphs.

Among the plethora of works on domination in graphs, and variants thereof, we briefly discuss the work more directly related to ours. The algorithmic aspect of total vector domination in strongly chordal graphs (a super-class of trees) was studied in [7], where polynomial time algorithms for that purpose were given. However, the authors of [7] point out that their approach cannot be modified to handle the case of vector domination, and that a new approach is needed.

Strictly related to our results is also the paper [12]. The authors study the hardness of approximating minimum monopolies in graphs. Monopolies in graphs represent an important sub-area in graph theory, with many applications in distributed computing (see the survey [15]). In our language, a monopoly corresponds to a total $1/2$-dominating set, and a partial monopoly to a $1/2$-dominating set. Therefore, our inapproximability results for q-domination and total q-domination can be seen as extensions of the results of [12] from the case $q = 1/2$ to arbitrary q (the paper [12] obtains better inapproximability multiplicative constants under stronger complexity assumptions). Conversely, our algorithmic results on trees extend the corresponding result of [13] from total $1/2$-domination to the general vector domination problem.

The paper [6] studies the hardness of approximating k-tuple domination in graphs. In our framework, this is equivalent to vector domination in which the input vector $(k_v : v \in V)$ has all components equal to integer k. The paper [11] studies the problem of bounding the size of a minimum-cardinality k-tuple total dominating set.

Our results are also somewhat related to the important new area of influence spread in social networks [5]. In particular, the paper [18] introduced the problem of identifying a minimum set of nodes that could influence a whole network within a time bound d. There, a set of nodes S influences a new node x in one step ($d = 1$) if the majority of neighbors of x is in S. The paper [18] mostly studies hardness results for the case $d = 1$. It is clear that our scenario corresponds to a much more general model of influence among nodes, similar to the one considered in [14] for a related but different problem.

2 Approximability results

In this section, we show that vector domination and total vector domination can be approximated in polynomial time by a factor of $H_{2\Delta(G)}$ and $H_{\Delta(G)}$, respectively. (We denote by $H_k = \sum_{i=1}^{k} \frac{1}{i}$ the k-th harmonic number.) Since $H_k \leq \log k + 1$ for $k \geq 1$, the algorithms given by the theorems below provide $O(\log \Delta(G))$-approximation for vector and total vector domination, respectively. (We denote by $\log$ the natural logarithm.)

Theorem 1 *Vector domination can be approximated in polynomial time by a factor of $H_{2\Delta(G)}$.*

Proof. For a graph $G = (V, E)$ and a vector $(k_v : v \in V)$ s.t. for all $v \in V$, $k_v \in \{0, 1, \ldots, d(v)\}$, we define a function $f : 2^V \to \mathbb{N}$, as follows:

$$f(S) = \sum_{v \in V} \tau_v(S)\,, \quad \text{where } \tau_v(S) = \begin{cases} \min\{|S \cap N(v)|, k_v\}, & \text{if } v \notin S; \\ k_v, & \text{if } v \in S. \end{cases} \tag{1}$$

The following properties of f can be verified: (i) f is integer-valued; (ii) $f(\emptyset) = 0$; (iii) f *is non-decreasing*; (iv) *A set $S \subseteq V$ satisfies $f(S) = f(V)$ if and only if S is a vector dominating set*; (v) f *is submodular.* (Recall that a real-valued set function f defined on the power set of a finite ground set R is called *submodular* if $f(X)+f(Y) \geq f(X \cup Y)+f(X \cap Y)$, for any two sets $X, Y \subseteq R$.) The only non-trivial result is (v), i.e, the submodularity of f. The proof is given below.

Fact 1 *Let $G = (V, E)$ be a graph and let $(k_v : v \in V)$ be a vector such that for all $v \in V$, $k_v \in \{0, 1, \ldots, d(v)\}$. The function $f : 2^V \to \mathbb{N}$, given by $f(S) = \sum_{v \in V} \tau_v(S)$, where $\tau_v(S)$ is defined in (1), is submodular.*

Proof. It suffices to show that all the functions $\tau_v(\cdot)$ are submodular. We complete the proof by verifying that τ_v satisfies the following characteristic property of submodular functions:

For all $S \subseteq T \subseteq V$ and for all $w \in V$,

$$\tau_v(T \cup \{w\}) - \tau_v(T) \leq \tau_v(S \cup \{w\}) - \tau_v(S)\,. \tag{2}$$

Observe that τ_v is non-decreasing.

Suppose first that $\tau_v(T) = k_v$. Then $\tau_v(T \cup \{w\}) = k_v$ and the left-hand side of inequality (2) is equal to 0. Hence inequality (2) holds since τ_v is non-decreasing.

From now on, we assume that $\tau_v(T) < k_v$, which implies $\tau_v(T) = |T \cap N_G(v)|$. Since τ_v is non-decreasing, $\tau_v(S) < k_v$, and hence $\tau_v(S) = |S \cap N_G(v)|$. Inequality (2) simplifies to

$$\tau_v(T) - \tau_v(S) = |(T \setminus S) \cap N_G(v)| \geq \tau_v(T \cup \{w\}) - \tau_v(S \cup \{w\})\,. \tag{3}$$

We may assume that $\tau_v(T \cup \{w\}) > \tau_v(S \cup \{w\})$, since otherwise the right-hand side of (3) equals 0, and inequality (3) holds.

Therefore, $\tau_v(S \cup \{w\}) < k_v$, implying $\tau_v(S \cup \{w\}) = |(S \cup \{w\}) \cap N_G(v)|$. If also $\tau_v(T \cup \{w\}) < k_v$ then $\tau_v(T \cup \{w\}) = |(T \cup \{w\}) \cap N_G(v)|$ and equality holds in (3).

So we may assume that $\tau_v(T \cup \{w\}) = k_v$. Note that v does not belong to $T \cup \{w\}$ (since otherwise either $\tau_v(T)$ or $\tau_v(S \cup \{w\})$ would equal to k_v). Suppose that the inequality (3) fails. Then

$$|(T \setminus S) \cap N_G(v)| < k_v - |(S \cup \{w\}) \cap N_G(v)|\,,$$

which implies

$$|(T \cup \{w\}) \cap N_G(v)| < k_v\,.$$

However, together with the fact that $v \notin T \cup \{w\}$, this contradicts the assumption that $\tau_v(T \cup \{w\}) = k_v$. $\square$

Back to the proof of Theorem 1, by (iv) we have that an optimal solution to the vector dominating set is provided by a minimum size S such that $f(S) = f(V)$. In other words, we have recast vector domination as a particular case of the well known `MINIMUM SUBMODULAR COVER` [17].

Let $\mathbb{A}$ denote the natural greedy strategy which starts with $S = \emptyset$ and iteratively adds to S the element $v \in V \setminus S$ s.t. $f(S \cup \{v\}) - f(S)$ is maximum, until $f(S) = f(V)$ is achieved.

By a classical result of Wolsey [17], it follows that algorithm $\mathbb{A}$ is an H_τ-approximation algorithm for vector domination, where $\tau = \max_{y\in V} f(\{y\})$. For every $y \in V$, we have $f(\{y\}) = \sum_{v\in V\setminus\{y\}} \tau_v(\{y\}) + \tau_y(\{y\}) \leq d(y) + k_y \leq 2d(y)$. Hence $\max_{y\in V} f(\{y\}) \leq 2\Delta(G)$ yielding the desired result. □

Theorem 2 *Total vector domination (and hence q-domination) can be approximated in polynomial time by a factor of $H_{\Delta(G)}$.*

Proof. The argument is analogous to the one used for Theorem 1. Given a graph $G = (V, E)$ and vector $(k_v : v \in V)$ s.t. for all $v \in V$, $k_v \in \{0, 1, \dots, d(v)\}$, we define a function $f : 2^V \to \mathbb{N}$, as follows:

$$f(S) = \sum_{v\in V} \min\{|S \cap N(v)|, k_v\}\,. \tag{4}$$

Like (1) also this function f satisfies property (i)-(iii) of the previous result. Moreover, a set $S \subseteq V$ satisfies $f(S) = f(V)$ if and only if S is a *total* vector dominating set. Finally f is submodular, as we will prove below.

Fact 2 *Let $G = (V, E)$ be a graph and let $(k_v : v \in V)$ be a vector such that for all $v \in V$, $k_v \in \{0, 1, \dots, d(v)\}$. The function $f : 2^V \to \mathbb{N}$, given by*

$$f(S) = \sum_{v\in V} \min\{|S \cap N_G(v)|, k_v\}\,,$$

is submodular.

Proof. For $v \in V$ and $S \subseteq V$, we set $\tau_v(S) = \min\{|S \cap N(v)|, k_v\}$. Since f is defined as the sum of the functions τ_v, and the sum of submodular functions is submodular, it suffices to show that the functions τ_v are submodular. For an arbitrary real number r and an arbitrary submodular function $g : 2^V \to \mathbb{R}$, the function $g' : 2^V \to \mathbb{R}$ defined by $g'(S) = \min\{g(S), r\}$ is submodular. Hence it suffices to show that the function $g_v(S) = |S \cap N(v)|$ is submodular. But this is easily seen to be the case since g_v can be written as the sum $g_v(S) = \sum_{u\in N(v)} \chi_u(S)$, where $\chi_u(S)$ is the (submodular) function taking value 1 if $u \in S$, and 0 otherwise. Therefore, f is submodular too. □

Therefore, again by the results of Wolsey [17] the natural greedy strategy provides an H_τ-approximation algorithm for total vector domination, where $\tau = \max_{y\in V} f(\{y\})$. It can be seen that $\max_{y\in V} f(\{y\}) \leq \Delta(G)$, which concludes the proof. □

3 Inapproximability results

Recall the following result on the inapproximability of domination and total domination, which was derived from the analogous result about the set cover problem due to Feige [4].

Theorem 3 *[1] For every $\epsilon > 0$, there is no polynomial time algorithm approximating domination (total domination) within a factor of $(1-\epsilon)\log n$, unless NP $\subseteq$ DTIME$(n^{O(\log\log n)})$.*

Our inapproximability results are given in terms of the q-domination problem. In fact, it turns out that both the q-domination and its total variant are inapproximable within a $\log |V(G)|$ factor as shown in Theorems 4 and 5 below. *A fortiori* the same results hold for the vector domination problem. Hence the approximations results of the previous section are basically best possible. We shall use the following lemma which is basically an *ad hoc* extension of the hardness of approximating domination within $c\log |V(G)|$ given in [1].

Lemma 1 *There exists a constant $c > 0$ such that for every integer $B > 0$ there is no polynomial time algorithm approximating domination on input graphs G satisfying $\gamma(G) \geq B\Delta(G)$ within a factor of $c\log |V(G)|$, unless NP $\subseteq$ DTIME$(n^{O(\log\log n)})$.*

Proof. Let B be a positive integer. We make a reduction from domination on general graphs. Let G be a graph with $|V(G)| \geq B$ that is an instance to domination. We transform G into a graph G' which consists of $r = B\Delta(G)$ disjoint copies of G, say $G_1, \ldots, G_r$. Then clearly $\gamma(G') = r\gamma(G)$, while $\Delta(G') = \Delta(G)$. In particular, since $\gamma(G) \geq 1$, the graph G' satisfies $\gamma(G') \geq r = B\Delta(G')$.

Let $\epsilon \in (0,1)$, and let $c = (1-\epsilon)/3$. For brevity, let us write $n = |V(G)|$ and $n' = |V(G')|$. Suppose that there exists an algorithm A that computes a $c\log n'$-approximation to domination in G'. Let S' be the set computed by A. Then $|S'| \leq c(\log n')\gamma(G')$.

For $i = 1, \ldots, r$, let $S'_i = S' \cap V(G_i)$, and let $S = S'_{i^*}$ such that $|S'_i| \leq |S'_{i^*}|$ for all $1 \leq i \leq r$. Then S is a dominating set in (the i^*-th copy of) G. Moreover, we can bound the size of S from above as follows:

$$
\begin{aligned}
|S| &\le (1/r)\cdot |S'| && \text{(by the definition of } S\text{)}\\
&\le (1/r)\cdot c(\log n')\cdot \gamma(G') && \text{(by the assumption on } A\text{)}\\
&\le (1/r)\cdot c(\log(rn))\cdot r\gamma(G) && \text{(by the properties of } G'\text{)}\\
&= c\log(rn)\gamma(G)\\
&\le c\log(n^3) && \text{(since } r\le n^2\text{)}\\
&= 3c\log n\\
&= (1-\epsilon)\log n\,.
\end{aligned}
$$

This shows that there is no polynomial time algorithm approximating domination on input graphs G satisfying $\gamma(G) \ge B\Delta(G)$ within a factor of $c\log|V(G)|$, unless NP $\subseteq$ DTIME($n^{O(\log\log n)}$). □

Theorem 4 *There exists a constant $c > 0$ such that for every $0 \le q < 1$ there is no polynomial time algorithm approximating q-domination within a factor of $c\log|V(G)|$, unless NP $\subseteq$ DTIME($n^{O(\log\log n)}$).*

Proof. Let $0 \le q < 1$, and let $B = \lceil \frac{q}{1-q} \rceil$. Let G be a graph with $\gamma(G) \ge B\Delta(G)$ and such that $|V(G)| \ge 4B$. We transform G into a graph G' which consists of G together with a complete graph K on $k = B\Delta(G)$ vertices such that K is disjoint from G. In addition, every vertex v from G is adjacent to precisely $k_v = \lfloor \frac{qd_G(v)}{1-q} \rfloor$ vertices in K. (This assignment is done in an arbitrary way.) Finally, every vertex in the clique K has a private neighbor outside $V(G)$. We denote by X the set of all these private neighbors. (See Fig. 1.)

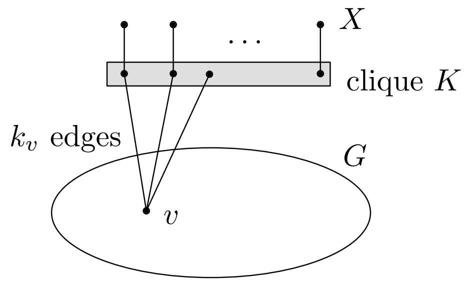

Fig. 1. The graph G' in the proof of Theorem 4

Notice that $k_v = \left\lfloor \frac{qd_G(v)}{1-q} \right\rfloor \le \left\lceil \frac{qd_G(v)}{1-q} \right\rceil \le \left\lceil \frac{q}{1-q} \right\rceil \Delta(G) = k$. Hence it is indeed possible to assign to v precisely k_v neighbors in K.

In addition, k_v is an integer satisfying $\frac{k_v}{d_G(v)+k_v} \leq q < \frac{k_v+1}{k_v+d_G(v)}$, which are instrumental to the following result.

Claim: $\gamma^q(G') = \gamma(G) + k$.

Proof. Let S' be an optimal q-dominating set in G'. Then, the set $S := S' \cap V(G)$ is a dominating set in G. Indeed, suppose for contradiction that there exists a vertex v in G such that S misses the closed neighborhood of v. Then $|N_{G'}(v) \cap S'| \leq k_v$. The degree of v in G' is equal to $d_{G'}(v) = d_G(v) + k_v$. Therefore

$$\frac{|N_{G'}(v) \cap S'|}{d_{G'}(v)} \leq \frac{k_v}{d_G(v) + k_v} \leq q\,,$$

contrary to the assumption that S' is q-dominating. Notice also that $|S' \backslash V(G)| \geq k$ since S' must meet each of the k edges connecting X to K. This implies that $|S| \leq |S'| - k$ and consequently $\gamma(G) \leq \gamma^q(G') - k$.

Conversely, let S be an optimal dominating set in G. The set $S' := S \cup K$ is then a q-dominating set in G' such that $|S'| = \gamma(G) + k$. To see that S' is q-dominating in G', observe that:

- For every $v \in V(G) \backslash S'$, the set $N_{G'}(v) \cap S'$ is the disjoint union of sets $N_G(v) \cap S$ and $N_{G'}(v) \cap K$. Hence

$$|N_{G'}(v) \cap S'| = |N_G(v) \cap S| + |N_{G'}(v) \cap K| \geq 1 + k_v >$$

$$> q(d_G(v) + k_v) = q d_{G'}(v)\,.$$

  The second inequality holds by the choice of k_v.
- Every $v \in K$ is contained in S'.
- For every $v \in X$, the unique neighbor of v is contained in S' and therefore $|N_{G'}(v) \cap S'| = 1 > q = q d_{G'}(v)$.

This shows that $\gamma^q(G') \leq \gamma(G) + k$ and completes the proof of the claim.

Let c be the constant given by Lemma 1, and let $c' = c/4$. Again, let us write $n = |V(G)|$ and $n' = |V(G')|$. Note that, by the assumption $|V(G)| \geq 4B$, it follows that $n' = n + 2k = n + 2B\Delta(G) \leq 1/2n^2 + 1/2n^2 = n^2$. Suppose, by contradiction, that there exists an algorithm A which computes a q-dominating set S' for G' such that $|S'| \leq c'(\log n')\gamma^q(G')$.

Let $S = S' \cap V(G)$. It is not hard to see that S is a dominating set in G. Moreover, we can bound the size of S as follows:

$$|S| \leq |S'| \leq c'(\log n')\gamma^q(G') \leq c'(\log(n^2))(\gamma(G) + k) \leq 2c'(\log n)(2\gamma(G)),$$

where the second inequality follows by the assumption on A; the third one by $n' \leq n^2$ and $\gamma^q(G') = \gamma(G) + k$; the fourth one by $k = B\Delta(G) \leq \gamma(G)$. Finally, by the choice of c' we get $|S| \leq c(\log n)\gamma(G)$, and the conclusion follows by Lemma 1. □

By means of a slightly more involved construction, we now prove the analogous result for total q-domination.

Theorem 5 *There exists a constant $c > 0$ such that for every $0 \leq q < 1$, there is no polynomial time algorithm approximating total q-domination within a factor of $c \log |V(G)|$, unless NP $\subseteq$ DTIME$(n^{O(\log \log n)})$.*

Proof. Let $0 \leq q < 1$, and let $B = \lceil \frac{q}{1-q} \rceil$. We make a reduction from total domination on graphs G satisfying $\gamma_t(G) \geq \max\{3, B, 2/q\}$.

Let G be a graph satisfying

$$\gamma_t(G) \geq \max\{3, B, 2/q\} . \tag{5}$$

Let $n = |V(G)|$. We transform G into a graph G' as follows: G' consists of n^3 disjoint copies of G, say $G_1, \ldots, G_{n^3}$, together with a complete graph K on Bn^3 vertices such that K is disjoint from the n^3 copies of G. Every vertex in the clique K also has a private neighbor outside the copies od G. We denote by X the set of all these private neighbors. (See Fig. 2.) To describe the remaining edges, we first partition the vertex set of K into n^2 equally-sized parts $K_1, \ldots, K_{n^2}$. (In particular, $|K_i| = Bn$ for all $i = 1, \ldots, n^2$.) Finally, for every $j \in \{1, \ldots, n^3\}$, we make every vertex $v \in V(G_j)$ adjacent to precisely k_v vertices in $K_{\lceil j/n \rceil}$ where k_v is an integer satisfying

$$\frac{k_v}{d_G(v) + k_v} \leq q < \frac{k_v + 1}{k_v + d_G(v)} .$$

Similarly as in the proof of Theorem 4, observe that we can take $k_v = \lfloor \frac{q d_G(v)}{1-q} \rfloor$. Also notice that since $k_v \leq Bn$, it is indeed possible to assign to every $v \in V(G_j)$ precisely k_v neighbors in $K_{\lceil j/n \rceil}$. (This assignment is done in an arbitrary way.)

Claim: $\gamma_t^q(G') = n^3\gamma_t(G) + Bn^3$.

Proof of Claim:

Let S' be an optimal total q-dominating set in G', that is, $|S'| = \gamma_t^q(G')$. For every $j = 1, \ldots, n^3$, let $S'_j = S' \cap V(G_j)$ denote the part of S' that belongs to to the j-th copy of G in G'. Pick an index $j^* \in \{1, \ldots, n^3\}$ for which the value of $|S'_j|$ is the smallest.

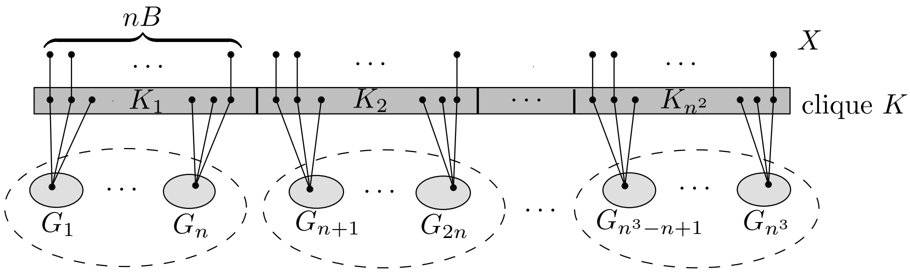

Fig. 2. The graph G' in the proof of Theorem 5

First, we argue that the set $S := S_{j^*}$ is a total dominating set in G_{j^*} (and thus in G). Indeed, suppose for contradiction that there exists a vertex v in G_{j^*} such that S misses the neighborhood of v. Then $|N_{G'}(v) \cap S'| \leq k_v$ while the degree of v in G' is equal to $d_{G'}(v) = d_G(v) + k_v$. Therefore

$$\frac{|N_{G'}(v) \cap S'|}{d_{G'}(v)} \leq \frac{k_v}{d_G(v) + k_v} \leq q\,,$$

contrary to the assumption that S' is total q-dominating. This implies that $\gamma_t(G) \leq |S|$.

Notice that $K \subseteq S'$ since otherwise there would exist a vertex $v \in X$ which would not be q-dominated by S'. Therefore,

$$|S'| \geq \sum_{j=1}^{n^3} |S'_j| + |K| \geq n^3|S| + Bn^3 \geq n^3\gamma_t(G) + Bn^3\,,$$

which shows that $\gamma_t^q(G') \geq n^3\gamma_t(G) + Bn^3$.

Conversely, let S be an optimal total dominating set in G. For $j = 1, \ldots, n^3$, let S_j denote the copy of S in G_j, and let $S' = K \cup \bigcup_{j=1}^{n^3} S_j$. The set $S' \subseteq V(G')$ satisfies $|S'| = n^3\gamma_t(G) + Bn^3$. Moreover, S' is a total q-dominating set in G':

- For every $j = 1, \ldots, n^3$ and for every $v \in V(G_j)$, the set $N_{G'}(v) \cap S'$ is the disjoint union of sets $N_{G_j}(v) \cap S_j$ and $N_{G'}(v) \cap K$. Hence

  $$|N_{G'}(v) \cap S'| = |N_{G_j}(v) \cap S_j| + |N_{G'}(v) \cap K| \geq 1 + k_v > q(d_{G_j}(v) + k_v) = q d_{G'}(v)\,.$$

  The second inequality holds by the choice of k_v.

- Let $v \in K$. By construction of G', v is adjacent to every other vertex in K, to precisely one vertex in X and to at most n^2 remaining vertices. Hence $d_{G'}(v) \leq (|K|-1)+1+n^2 = Bn^3+n^2$. Moreover, $|N_{G'}(v) \cap S'| \geq |K|-1 = Bn^3-1$. Then

$$\frac{|N_{G'}(v) \cap S'|}{d_{G'}(v)} \geq \frac{Bn^3-1}{Bn^3+n^2} > q\,.$$

  Indeed, the inequality $\frac{Bn^3-1}{Bn^3+n^2} > q$ is equivalent to the inequality $B(1-q)n^3 > 1+qn^2$, which holds true since

$$Bn^3(1-q) \geq qn^3 \geq 2n^2 > 1+qn^2\,,$$

  as can be seen using the fact that $B \geq \frac{q}{1-q}$ and the assumption that $n \geq \gamma_t(G) \geq 2/q$.
- For every $v \in X$, the unique neighbor of v is contained in S' and therefore $|N_{G'}(v) \cap S'| = 1 > q = qd_{G'}(v)$.

This shows that $\gamma_t^q(G') \leq n^3\gamma_t(G)+Bn^3$ and completes the proof of the claim.

Let $\epsilon \in (0,1)$ and let $c = (1-\epsilon)/10$.

Let us write $n' = |V(G')| = n^4+2Bn^3$. By Assumption (5) we have $n \geq B$ and $n \geq 3$ and hence

$$n' = n^4+2Bn^3 \leq n^4+2n^4 = 3n^4 \leq n^5\,.$$

Suppose that there exists an algorithm A that computes a $c\log n'$-approximation to total q-domination in G'. Let S' be the set computed by A. Then $|S'| \leq c(\log n')\gamma_t^q(G')$.

Similarly as in the proof of the claim above, let $S'_j = S' \cap V(G_j)$ and pick an index $j^* \in \{1,\ldots,n^3\}$ for which the value of $|S'_j|$ is the smallest. Then, setting $S = S'_{j^*}$ results in a total dominating set in G_j (and hence in G).

We can bound the size of S from above as follows:

$$\begin{aligned}
|S| &\leq \tfrac{1}{n^3}|S'| && \text{(by the choice of } j^*\text{)}\\
&\leq \tfrac{1}{n^3}c(\log n')\gamma_t^q(G') && \text{(by the assumption on } A\text{)}\\
&\leq \tfrac{1}{n^3}c(\log(n^5))(n^3\gamma_t(G)+Bn^3) && \text{(since } n' \leq n^5 \text{ and } \gamma_t^q(G') = n^3\gamma_t(G)+Bn^3\text{)}\\
&= 5c(\log n)(\gamma_t(G)+B) &&\\
&\leq 10c(\log n)\gamma_t(G) && \text{(since } B \leq \gamma_t(G)\text{)}\\
&= (1-\epsilon)(\log n)\gamma_t(G)\,. &&
\end{aligned}$$

Therefore, S approximates the total domination within a factor of $(1-\epsilon)\log n$. This shows that there is no polynomial time algorithm approximating total q-domination within a factor of $c\log|V(G)|$, unless NP $\subseteq$ DTIME$(n^{O(\log\log n)})$. $\square$

4 Polynomial algorithms for particular graph classes

4.1 Complete graphs

Proposition 1. *Let G be a complete graph with vertex set $V(G)=\{v_1,\ldots,v_n\}$ and assume that $n-1\geq k(v_1)\geq\cdots\geq k(v_n)\geq k(v_{n+1}):=0$ with $k(v_1)>0$. Then, a minimum vector dominating set for (G,k) is given by $D=\{v_1,\ldots,v_p\}$ where $p=\min\{i\,:\,1\leq i\leq n,\,i\geq k(v_{i+1})\}$.*

Proof. Clearly, $D=\{y_1,\ldots,v_p\}$ as above is a vector dominating set for (G,k) since every $v\in V(G)\setminus D$ is of the form $v=v_j$ for some $j\geq p+1$ and therefore $|N(v_j)\cap D|=|D|=p\geq k(v_{p+1})\geq k(v_j)$. Conversely, if D is a set of at most $p-1$ vertices, then there exists a vertex $v_i\in V(G)\setminus D$ such that $i\leq p$. By definition of p, we have $p-1<k(v_p)$. Therefore, $|N(v_i)\cap D|\leq p-1<k(v_p)\leq k(v_i)$, hence D is not a vector dominating set for (G,k). $\square$

Corollary 1 *For complete graphs, the vector domination problem is solvable in time $O(p\log n)$, where $p=\min\{i\,:\,1\leq i\leq n,\,i\geq k(v_{i+1})\}$.*

Total vector domination is even simpler, and solvable in $O(n)$ time.

Proposition 2. *Let $G=(V,E)$ be a complete graph. Let $K=\max\{k(v)\,:\,v\in V(G)\}$ and let $M=\{v\in V\,:\,k(v)=K\}$. If $|M|\leq|V|-K$, then a minimum total vector dominating set for (G,k) is given by any subset of K vertices contained in $V\setminus M$. Otherwise, a minimum total vector dominating set for (G,k) is given by any subset of $K+1$ vertices.*

4.2 Trees

Since trees are strongly chordal, total vector domination is solvable in time $O(n+m)$ on trees [7, 16], where $n=|V(G))|$ and $|E(G)|$. As mentioned in [7], their approach does not apply to the vector domination problem. In this section we describe an $O(n^2)$ algorithm that solves vector domination in trees. The algorithm is based on an efficient solution to the following problem:

CARDINALITY-CONSTRAINED PARTITION (*CCP*):
Given n ordered pairs of real numbers $(a_1, b_1), (a_2, b_2), \ldots, (a_n, b_n)$ and an integer k such that $0 \leq k \leq n$, find a partition (I, J) of the set $\{1, \ldots, n\}$ with $|I| = k$ that minimizes the sum $\sum_{i \in I} a_i + \sum_{j \in J} b_j$.

This problem admits an $O(n(k+1))$ solution by dynamic programming: For $1 \leq i \leq n$ and $0 \leq j \leq \min\{i, k\}$, let $v_{i,j}$ denote the optimum value of the *CCP* problem with the input $(a_1, \ldots, a_i, b_1, \ldots, b_i; j)$. Clearly, $v_{n,k}$ is an optimal value of the above *CCP* problem. The values $v_{i,j}$ can be computed in $O(n(k+1))$ time using the following straightforward recurrences:

- $v_{1,0} = b_1, \quad v_{1,1} = a_1$;
- $v_{i,0} = v_{i-1,0} + b_i$ for all $2 \leq i \leq n$;
- $v_{i,i} = v_{i-1,i-1} + a_i$ for all $2 \leq i \leq k$;
- $v_{i,j} = \min\{v_{i-1,j-1} + a_i, v_{i-1,j} + b_i\}$ for all $2 \leq i \leq k$ and $1 \leq j \leq \min\{i, k\}$.

As usual, an optimal solution can be obtained by a standard backtracking procedure.

In what follows, we will denote by *CCP*(A, k) the optimal value of the *CCP* problem on the input $((a_{11}, a_{21}), \ldots, (a_{n1}, a_{n2}); k)$, where $A = \begin{pmatrix} a_{11} & \cdots & a_{n1} \\ a_{21} & \cdots & a_{n2} \end{pmatrix}$ is a $2 \times n$ matrix and $k \leq n$ is a non-negative integer.

Theorem 6 *A minimum vector dominating set in a tree can be found in time $O(n^2)$.*

Proof. We claim that Algorithm 1 below computes a minimum vector dominating set for (T, k), where T is a tree. Let us root T at an arbitrary vertex r. For $v \in V(T)$, we denote by T_v the subtree of T induced by v and all its descendants. For a subgraph H of G, we denote by $k|_H$ the restriction of k to $V(H)$. The algorithm will compute, using a bottom-up traversal of the tree, the following values, for all $v \in V(T)$:

- $\gamma(v)$: the minimum size of a vector dominating set for $(T_v, k|_{T_v})$;
- $\gamma^+(v)$: the minimum size of a vector dominating set for $(T_v, k|_{T_v})$ that contains v;
- $\gamma^-(v)$: the minimum size of a vector dominating set for (T_v, k_v^-), where $k_v^- : V(T_v) \to \mathbb{Z}$ is given by $k_v^-(u) = \begin{cases} \max\{k(v) - 1, 0\}, & \text{if } u = v; \\ k(u), & \text{otherwise.} \end{cases}$

The following proposition establishes a way to compute these values:

Proposition 3. *Let v be an internal node of T, and let $C(v)$ denote the set of children of v. Let A be the $2 \times |C(v)|$ matrix with $A_{1j} = \gamma^+(j)$ and $A_{2j} = \gamma(j)$ for all $j \in C(v)$. Then:*

(i) $\gamma^+(v) = \sum_{j \in C(v)} \gamma^-(j) + 1$.

(ii) If $k(v) > |C(v)|$ then $\gamma(v) = \gamma^+(v)$. Otherwise, $\gamma(v) = \min\{\gamma^+(v), CCP(A, k(v))\}$.

(iii) If $k(v) > |C(v)| + 1$ then $\gamma^-(v) = \gamma^+(v)$. Otherwise, $\gamma^-(v) = \min\{\gamma^+(v), CCP(A, \max\{k(v) - 1, 0\})\}$.

Proof. (i). For all $j \in C(v)$, let D_j denote a minimum vector dominating set for (T_j, k_j^-). Then, the set $\cup_j D_j \cup \{v\}$ is a vector dominating set for $(T_v, k|_{T_v})$ that contains v; hence $\gamma^+(v) \leq \sum_{j \in C(v)} \gamma^-(j) + 1$. Conversely, if D is a minimum-sized vector dominating set for $(T_v, k|_{T_v})$ that contains v, then for every $j \in C(v)$, the set $D \cap V(T_j)$ is a vector dominating set for (T_j, k_j^-). Therefore, $|D \cap V(T_j)| \geq \gamma^-(j)$ and consequently $\gamma^+(v) = |D| \geq \sum_{j \in C(v)} \gamma^-(j) + 1$.

(ii). If $k(v) > |C(v)|$ then every vector dominating set for $(T_v, k|_{T_v})$ contains v, so we have $\gamma(v) = \gamma^+(v)$ in this case. Suppose now that $k(v) \leq |C(v)|$. First, we show the inequality "$\leq$". It follows from the definitions that $\gamma(v) \leq \gamma^+(v)$. Let (I, J) be an optimal solution for $CCP(A, k(v))\}$. For all $j \in C(v)$ define

$$D_j = \begin{cases} \text{a minimum vector dominating set for } (T_j, k|_{T_j}) \text{ that contains } j, & \text{if } j \in I\ ; \\ \text{a minimum vector dominating set for } (T_j, k|_{T_j}), & \text{otherwise.} \end{cases}$$

Then, $j \in D_j$ for all $j \in I$. Therefore, since $|I| = k(v)$, the set $D := \cup_{j \in C(v)} D_j$ is a vector dominating set for $(T_v, k|_{T_v})$. Consequently, $\gamma(v) \leq CCP(A, k(v))$.

To see the converse inequality, suppose that $\gamma(v) < \gamma^+(v)$ (otherwise, we are done). For every minimum vector dominating set D for $(T_v, k|_{T_v})$ it holds that $v \notin D$ and also $|D \cap C(v)| \geq k(v)$. Hence, it is enough to show that

$$\gamma(v) \geq \min_{\substack{I \subseteq C(v) \\ |I| = k(v)}} \left(\sum_{i\ \in I} \gamma^+(i) + \sum_{i\ \in C(v) \setminus I} \gamma(i) \right). \tag{6}$$

Let D be a minimum vector dominating set for $(T_v, k|_{T_v})$ and let $I \subseteq D \cap C(v)$ such that $|I| = k(v)$. Then, for all $i \in I$, the set $D \cap V(T_i)$ is a minimum-sized vector dominating set for $(T_i, k|_{T_i})$ that contains i (otherwise, a smaller such set, say D'_i, could be used to produce a vector dominating set for $(T_v, k|_{T_v})$ smaller than D – namely $(D \setminus V(T_i)) \cup D'_i$). Therefore $|D \cap V(T_i)| = \gamma^+(i)$. Similarly, $|D \cap V(T_i)| = \gamma(i)$ for all $i \in C(v) \setminus I$. Summing up over all i, we get $\sum_{i \in I} \gamma^+(i) + \sum_{i \in C(v) \setminus I} \gamma(i) = \sum_{i \in C(v)} |D \cap V(T_i)| = |D| = \gamma(v)$. Inequality (6) follows.

The proof of (iii) is similar to that of (ii). □

Algorithm 1 Vector domination in trees

Input: A tree $T = (V, E)$, a function $k : V \to \mathbb{Z}$.
Output: The minimum size of a vector dominating set for (T, k).

Let $R = \{v \in V(T) \ : k(v) > d(v)\}$.
Set T to $T - R$ and k to $k' : V(T - R) \to \mathbb{Z}$, given by $k'(v) = k(v) - |N(v) \cap R|$ for all $v \in V(T) - R$.
if $T - R$ is disconnected, with components $T_1, \dots, T_p$
    Solve the problem recursively on all $(T_i, k|_{T_i})$, for all i.
    Let γ_i denote the minimum size of a vector dominating set for $(T_i, k|_{T_i})$.
    return $\sum_{i=1}^{p} \gamma_i + |R|$.
Let $r \in V(T)$ and root T at r.
for all leaves ℓ of T (other than the root) **do**
    set $\gamma(\ell) = k(\ell)$. (since at this point $k(\ell) \in \{0, 1\}$.)
    set $\gamma^+(\ell) = 1$.
    set $\gamma^-(\ell) = 0$.
for all internal nodes v of T (traversed in a bottom-up manner) **do**
    let $C(v)$ be the set of children of v.
    set $\gamma^+(v) = \sum_{j \in C(v)} \gamma^-(j) + 1$.
    let A be the $2 \times |C(v)|$ matrix with $A_{1j} = \gamma^+(j)$ and $A_{2j} = \gamma(j)$ for all $j \in C(v)$.
    if $k(v) > |C(v)|$
        set $\gamma(v) = \gamma^+(v)$.
    else
        set $\gamma(v) = \min\{\gamma^+(v), \mathit{CCP}(A, k(v))\}$.
    if $k(v) > |C(v)| + 1$
        set $\gamma^-(v) = \gamma^+(v)$.
    else
        set $\gamma^-(v) = \min\{\gamma^+(v), \mathit{CCP}(A, \max\{k(v) - 1, 0\})\}$.
return $\gamma(r)$.

The correctness of the procedure follows from Proposition 3.

To analyze the time complexity, observe that at each leaf, a constant amount of computation is performed. The total time spent at an internal node v is proportional to $O(|C(v)|(k(v) + 1)) = O(d(v)^2)$. Altogether, this results in the time complexity of $O(n^2)$. Finally, an optimal solution can be computed with a standard backtracking procedure, via a top-down traversal of the tree. □

4.3 P_4-free graphs

In this section we give a polynomial time algorithm to solve the vector domination and total vector domination problems in P_4-free graphs. P_4-free graphs (also known as cographs) are graphs without an induced subgraph isomorphic to a 4-vertex path. A polynomial-time algorithm for the vector domination and total vector domination problems in P_4-free graphs can be developed based on the following well-known characterization of P_4-free graphs [2]: a graph G is P_4-free if and only if for every induced subgraph F of G with at least two vertices, either F or its complement is disconnected. A *co-component* of a graph $G = (V, E)$ is the subgraph of G induced by the vertex set of a connected component of the complementary graph $\overline{G} = (V, \{uv \mid u, v \in V,\ u \neq v,\ uv \notin E\})$. The above characterization implies that every P_4-free graph $G = (V, E)$ admits a recursive decomposition into one-vertex graphs by taking components or co-components. Such a decomposition can be computed in linear time [3], and a tree representing such a decomposition is called a *cotree*. For our purposes, it will be more convenient to assume that G is represented by a *modified cotree*, which is obtained from the cotree by replacing every node representing a decomposition of an induced subgraph F of G into $p \geq 3$ co-components $C_1, \ldots, C_p$ with $p - 1$ nodes in sequence, with i-th node representing the decomposition of $F_i := F - (C_1 \cup \cdots \cup C_{i-1})$ into C_i and $F_i - C_i$.

Proposition 4. *Let G, G_1, G_2 be graphs such that G is obtained from the disjoint union of G_1 and G_2 by adding all edges of the form $\{uv\ : u \in V(G_1),\ v \in V(G_2)\}$. Then,*

$$\gamma(G, k) = \min_{\substack{0 \leq i \leq |V(G_2)| \\ 0 \leq j \leq |V(G_1)|}} \big(\max\{\gamma(G_1, k_i), j\} + \max\{\gamma(G_2, k'_j), i\}\big)$$

$$\gamma_t(G, k) = \min_{\substack{0 \leq i \leq |V(G_2)| \\ 0 \leq j \leq |V(G_1)|}} \big(\max\{\gamma_t(G_1, k_i), j\} + \max\{\gamma_t(G_2, k'_j), i\}\big)\ ,$$

where $k_i(v) = \max\{k(v) - i, 0\}$ for all $v \in V(G_1)$ and $k'_j(v) = \max\{k(v) - j, 0\}$ for all $v \in V(G_2)$.

Proof. Let m denote the value of the first minimum above. First, we show that $m \leq \gamma(G, k)$. Let D be a minimum vector dominating set for (G, k), that is, $|D| = \gamma(G, k)$. Let $D_i = D \cap V(G_i)$, for $i = 1, 2$, and let $i^* = |D_2|$ and $j^* = |D_1|$. Take a vertex $v \in V(G_1) \setminus D_1$ such that $k_{i^*}(v) > 0$. Then

$$|N_{G_1}(v) \cap D_1| = |N_G(v) \cap D| - |D_2| = |N_G(v) \cap D| - i^* \geq k(v) - i^* = k_{i^*}(v)\,.$$

Therefore D_1 is a vector dominating set for (G_1, k_{i^*}) and consequently $\gamma(G_1, k_{i^*}) \le |D_1| = j^*$. Similarly, we can show that $\gamma(G_2, k'_{j^*}) \le |D_2| = i^*$. It follows that

$$\gamma(G,k) = |D| = j^* + i^* = \max\{\gamma(G_1, k_{i^*}), j^*\} + \max\{\gamma(G_2, k'_{j^*}), i^*\} \ge m\,.$$

To see the converse inequality, let (i^*, j^*) be a pair of indices where the value of m is attained. Let D_1 be a vector dominating set for (G_1, k_{i^*}) such that $|D_1| = \max\{\gamma(G_1, k_{i^*}), j^*\}$. Similarly, let D_2 be a vector dominating set for (G_2, k'_{j^*}) such that $|D_2| = \max\{\gamma(G_2, k_{j^*}), i^*\}$. Then, the set $D := D_1 \cup D_2$ is a vector dominating set for (G, k): Let $v \in V(G) \setminus D$. Assuming that $v \in V(G_1) \setminus D_1$, we have

$$|N_G(v) \cap D| = |N_{G_1}(v) \cap D_1| + |D_2| \ge k_{i^*}(v) + |D_2| \ge k(v) - i^* + |D_2| \ge k(v)\,.$$

We can show similarly that $|N_G(v) \cap D| \ge k(v)$ for all $v \in V(G_2) \setminus D_2$. Therefore, $\gamma(G,k) \le |D| = |D_1| + |D_2| = m$, which completes the proof.

The proof of the other relation is analogous. □

Theorem 7 *Vector domination problem and total vector domination problem are solvable in polynomial time on P_4-free graphs.*

Proof. We claim that Algorithm 2 below computes a minimum vector dominating set for (G, k), where G is a P_4-free graph. The following notations are used: For a non-negative integer r and for an induced subgraph H of G, we denote by $D(H, r)$ a minimum vector dominating set for (H, k_r), where $k_r(v) = \max\{k(v) - r, 0\}$ for all $v \in V(H)$.

In lines 1–2, the algorithm computes the set R of required vertices in every feasible solution, and reduces the problem to a smaller graph. Notice that once the required vertices have been removed, it holds that $k(v) \le d(v)$ for all v. In particular, for an induced subgraph H of the reduced graph $G - R$, it suffices to compute the sets $D(H, r)$ for $r \le \Delta(G)$, since $D(H, r') = \emptyset$ for every $r' \ge \Delta(G)$.

The correctness of the algorithm is straightforward, using the above-mentioned characterization of P_4-free graphs and Proposition 4 together with the arguments given in its proof. It is also easy to see that the algorithm runs in time $O(\Delta(G)n^3)$.

The algorithm can be modified slightly so that it computes a minimum total vector dominating set. Suppose that an induced subgraph H of G contains a

Algorithm 2 Vector domination in P_4-free graphs

Input: A P_4-free graph $G = (V, E)$, a function $k : V \to \mathbb{Z}$.
Output: A minimum vector dominating set for (G, k).

1: Let $R = \{v \in V(G) \ : k(v) > d(v)\}$.
2: Set G to $G - R$ and k to $k' : V(G - R) \to \mathbb{Z}$, given by $k'(v) = k(v) - |N(v) \cap R|$ for all $v \in V(G) - R$.
3: Compute a modified cotree T of G.
4: **for** all leaves ℓ of T **do**
5:     let $v \in V(G)$ be the vertex corresponding to ℓ.
6:     **for** $0 \leq r \leq \Delta(G)$ **do**
7:         set $D(\{v\}, r) = \begin{cases} \emptyset, & \text{if } k(v) \leq r; \\ \{v\}, & \text{otherwise.} \end{cases}$
8: **for** all internal nodes of T (traversed in a bottom-up manner) **do**
9:     let H be the subgraph of G corresponding to the current node of T.
10:     **if** H is disconnected, with connected components $C_1, \ldots, C_m$ **then**
11:         **for** $0 \leq r \leq \Delta(G)$ **do**
12:             set $D(H, r) = \cup_{1 \leq i \leq m} D(C_i, r)$.
13:     **else**
14:         let C be a co-component of H and let $H_2 = H - C$.
15:         **for** $0 \leq r \leq \Delta(G)$ **do**
16:             **for** $0 \leq i \leq |V(H_2)|$ **do**
17:                 let $D_i = D(C, \min\{r + i, \Delta(G)\})$.
18:             **for** $0 \leq j \leq |V(C)|$ **do**
19:                 let $D'_j = D(H_2, \min\{r + j, \Delta(G)\})$.
20:             let (i^*, j^*) be a pair of indices such that $\max\{|D_{i^*}|, j^*\} + \max\{|D'_{j^*}|, i^*\} = \min_{i,j} \left(\max\{|D_i|, j\} + \max\{|D'_j|, i\}\right)$.
21:             let $\hat{D}_1 = D_{i^*} \cup J$ where $J \subseteq V(C) \setminus D_{i^*}$ such that $|J| = \max\{j^* - |D_{i^*}|, 0\}$.
22:             let $\hat{D}_2 = D'_{j^*} \cup J$ where $J \subseteq V(G_2) \setminus D'_{j^*}$ such that $|J| = \max\{i^* - |D'_{j^*}|, 0\}$.
23:             set $D(H, r) = \hat{D}_1 \cup \hat{D}_2$.
24: **return** $D(G, 0) \cup R$.

vertex v such that $k(v) - r > d(v)$. In this case, we set $D(H, r) = \text{Inf}$ where Inf is a special symbol denoting the infeasibility of the problem (we also set $|\text{Inf}| = \infty$); moreover Inf is invariant under taking unions: $A \cup \text{Inf} = \text{Inf}$ for every A. We need the following modifications:

- replace lines 1–2 with the following:

  if there exists a vertex v such that $k(v) > d(v)$ **then** return Inf.
- replace line 7 with the following:

  set $D(\{v\}, r) = \begin{cases} \emptyset, & \text{if } k(v) \leq r; \\ \text{Inf}, & \text{otherwise.} \end{cases}$

□

4.4 Threshold graphs

Threshold graphs form a subclass of P_4-free graphs, therefore vector domination and total vector domination problems are solvable in polynomial time on threshold graphs. Since threshold graphs are strongly chordal, the total vector domination problem is solvable in time $O(n+m)$ on threshold graphs [7, 16]. We develop in this section an $O(n^2)$ algorithm for the vector domination problem in threshold graphs, using the following characterization: A graph $G = (V, E)$ is threshold if and only if there is an ordering $v_1, \dots, v_n$ of V such that for every i, vertex v_i is either isolated or dominating in the subgraph G_i of G induced by $\{v_1, \dots, v_i\}$. Such an ordering of a threshold graph G can be found in linear time by recursively removing dominating or isolated vertices.

We will also need the following proposition similar to Proposition 4. Recall that for a subgraph H of G, we denote by $k|_H$ the restriction of k to $V(H)$.

Proposition 5. *Let G be a graph with a dominating vertex v. Let $G' = G - \{v\}$ and $k' : V(G') \to \mathbb{Z}$ be given by $k'(u) = \max\{k(u) - 1, 0\}$ for all $u \in V(G')$. If $k(v) > d(v)$ then every minimum vector dominating set D for (G, k) is of the form $D' \cup \{v\}$ where D' is a minimum vector dominating set for (G', k'). Otherwise,*

$$\gamma(G,k) = \min\{\max\{\gamma(G', k|_{G'}), k(v)\}, 1 + \gamma(G', k')\}\,.$$

More specifically, if D' is a minimum vector dominating set for $(G', k|_{G'})$ then and D'' is a minimum vector dominating set for (G', k') then a minimum vector dominating set D for (G, k) can be computed as follows:

$$D = \begin{cases} D' \cup J, & \textit{if } \max\{|D'|, k(v)\} \leq 1 + \gamma(G', k'); \\ D'' \cup \{v\}, & \textit{otherwise,} \end{cases}$$

where $J \subseteq V(G') \setminus D'$ such that $|J| = \max\{k(v) - |D'|, 0\}$.

Proof. If $k(v) > d(v)$ then every minimum vector dominating set D for (G, k) must contain v, and the first statement follows.

Suppose now that $k(v) \leq d(v)$. Let D be a minimum vector dominating set for (G, k). If $v \in D$ then $D' = D \setminus \{v\}$ is a vector dominating set for (G', k'). Therefore, in this case $\gamma(G', k') \leq \gamma(G, k) - 1$ and the inequality $\gamma(G,k) \geq \min\{\max\{\gamma(G', k|_{G'}), k(v)\}, 1 + \gamma(G', k')\}$ follows. If $v \notin D$ then $D' = D \setminus \{v\}$ is a vector dominating set for $(G', k|_{G'})$, moreover $|D'| \geq k(v)$; therefore the inequality $\gamma(G,k) \geq \min\{\max\{\gamma(G', k|_{G'}), k(v)\}, 1+\gamma(G', k')\}$ holds in this case too.

To see the converse inequality, suppose first that $\max\{\gamma(G', k|_{G'}), k(v)\} \leq 1 + \gamma(G', k')$, and let D' be a minimum vector dominating set for $(G', k|_{G'})$. Let $D = D' \cup J$ where $J \subseteq V(G') \setminus D'$ such that $|J| = \max\{k(v) - |D'|, 0\}$. Then, the set D contains at least $k(v)$ neighbors of v, therefore D is a vector dominating set for (G, k). Similarly, if $\max\{\gamma(G', k|_{G'}), k(v)\} > 1+\gamma(G', k')$, then letting D'' be a minimum vector dominating set for (G', k'), we can define $D = D'' \cup \{v\}$ to obtain a vector dominating set for (G', k'). In summary, $\gamma(G, k) \leq \min\{\max\{\gamma(G', k|_{G'}), k(v)\}, 1 + \gamma(G', k')\}$; hence equality holds, and the set D is also a minimum vector dominating set for (G, k). $\square$

Proposition 5 leads to Algorithm 3 below for the vector domination problem on threshold graphs.

Algorithm 3 Vector domination in threshold graphs

Input: A threshold graph $G = (V, E)$, a function $k : V \to \mathbb{Z}$.
Output: A minimum vector dominating set for (G, k).

Let $R = \{v \in V(G)\ : k(v) > d(v)\}$.
Set G to $G - R$ and k to $k' : V(G - R) \to \mathbb{Z}$, given by $k'(v) = k(v) - |N(v) \cap R|$ for all $v \in V(G) - R$.
Compute an ordering $v_1, \ldots, v_n$ of $V(G)$ such that v_i is either isolated or dominating in G_i.
Compute the values p_j for all $j \in \{1, \ldots, n\}$.
for $0 \leq j \leq p_1$ **do**
  set $D_{1,j} = \begin{cases} \emptyset, & \text{if } k(v_1) \leq j; \\ \{v_1\}, & \text{otherwise.} \end{cases}$
for $i = 2, \ldots, n$ **do**
  if v_i is isolated in G_i
    for $0 \leq j \leq p_i$ **do**
      set $D_{i,j} = \begin{cases} D_{i-1,j}, & \text{if } k(v_i) \leq j; \\ D_{i-1,j} \cup \{v_i\}, & \text{otherwise.} \end{cases}$
  else
    for $0 \leq j \leq p_i$ **do**
      if $\max\{|D_{i-1,j}|, k(v) - j\} \leq 1 + |D_{i-1,j+1}|$
        let $J \subseteq V(G_{i-1}) \setminus D_{i-1,j}$ such that $|J| = \max\{k(v) - j - |D_{i-1,j}|, 0\}$.
        set $D_{i,j} = D_{i-1,j} \cup J$.
      else
        set $D_{i,j} = D_{i-1,j+1} \cup \{v_i\}$.
return $D_{n,0} \cup R$.

Theorem 8 *A minimum vector dominating set in a threshold graph can be found in time $O(nm)$.*

Proof. We claim that Algorithm 3 computes a minimum vector dominating set for (G, k), where G is a threshold graph. We use similar notation as in the proof of Theorem 7, except that we denote by $D_{i,j}$ a minimum vector dominating set for (G_i, k_j) where $k_j(v) = \max\{k(v) - j, 0\}$ for all $v \in V(G_i)$. The algorithm will compute, by dynamic programming, all sets $D_{i,j}$, for all $i \in \{1, \ldots, n\}$ and all $j \in \{0, 1, \ldots, p_i\}$ where p_i is the number of indices $j > i$ such that v_j is dominating in G_j.

The correctness of the algorithm follows by induction on i, using Proposition 5. Notice that for all $i \geq 2$ such that v_i is dominating in G_i, we have $p_{i-1} = p_i + 1$, therefore $j + 1 \leq p_{i-1}$ in lines 13 and 17, so $D_{i-1,j+1}$ has already been computed at that point. The total time complexity is $O(n \sum_{i=1}^{n} p_i) = O(nm)$, and can be improved to $O(n + m)$ if only the minimum size of a vector dominating set is needed. □

5 Concluding remarks

We have studied some algorithmic issues related to natural extensions of the well known concepts of domination and total domination in graphs. We have shown that the problems are approximable to within a logarithmic factor, and proved that this is essentially best possible. We have also provided exact polynomial time algorithms for several interesting classes of graphs, namely, complete graphs, trees, P_4-free graphs and threshold graphs.

References

1. M. CHLEBÍK and J. CHLEBÍKOVA. Approximation hardness of dominating set problems in bounded degree graphs. *Information and Computation* 206 (2008) 1264–1275.
2. D.G. CORNEIL, H. LERCHS and L. STEWART BURLINGHAM. Complement reducible graphs. *Discrete Appl. Math.* 3 (1981) 163–174.
3. D.G. CORNEIL, Y. PERL and L.K. STEWART. A linear recognition algorithm for cographs. *SIAM J. Comput.* 14 (1985) 926–934.
4. U. FEIGE. A threshold of $\ln n$ for approximating set cover. *Journal of ACM* 45 (1998) 634–652.
5. D. KEMPE, J.M. KLEINBERG, E. TARDOS. *Influential Nodes in a Diffusion Model for Social Networks.* ICALP 2005, 1127-1138
6. R.KLASING and C. LAFOREST. Hardness results and approximation algorithms of k-tuple domination in graphs. *Inf. Process. Lett.* 89(2): 75-83 (2004)
7. C.S. LIAO and G.J. CHANG. k-tuple domination in graphs. *Inform. Process. Lett.* 87 (2003) 45–50.
8. J. HARANT, A. PROCHNEWSKI, and M. VOIGT. On dominating sets and independent sets of graphs. *Combinatorics, Probability and Computing* 8 (1999), 547-553
9. T.W. HAYNES, S. HEDETNIEMI and P. SLATER *Fundamentals of Domination in Graphs.* Marcel Dekker, 1998.

10. T.W. HAYNES, S. HEDETNIEMI and P. SLATER (Eds.) *Domination in Graphs: Advanced Topics.* Marcel Dekker, 1998.
11. M.A. HENNING AND A.P. KAZEMI, k-tuple total domination in graphs. *Discrete Appl. Math.* 158 (2010) 1006–1011.
12. S. MISHRA, JAIKUMAR RADHAKRISHNAN and S. SIVASUBRAMANIAN. On the hardness of approximating minimum monopoly problems. *FST TCS 2002* LNCS Vol. 2556 2002, 277-288
13. S. MISHRA and S. B. RAO. Minimum monopolies in regular and trees graphs. *Discrete Mathematics* 306 (2006). 1586-1594.
14. E. MOSSEL, and S. ROCH. On the submodularity of influence in social networks. *Proc. 39th Ann. ACM Symp. on Theory of Comp.*, ACM, 2007, pp. 128–134.
15. D. PELEG. Local majorities, coalitions and monopolies in graphs: a review. *Theoretical Computer Science* 282 (2002) 231–257.
16. R. UEHARA. Linear time algorithms on chordal bipartite and strongly chordal graphs *Proc. ICALP 2002*, LNCS 2380, 993-1004, 2002.
17. L.A. WOLSEY. An analysis of the greedy algorithm for the submodular set covering problem. *Combinatorica* 2 (1982) 385–393.
18. F. ZOU, J. K WILLSON, Z. ZHANG, and W. WU. Fast information propagation in social networks. *Discrete Mathematics, Algorithms and Applications* 2 (2010) 1-17.