arXiv:1812.00689v1 [cs.DM] 3 Dec 2018

Total 2-domination of proper interval graphs

Francisco J. Soulignac*†

francisco.soulignac@unq.edu.ar

Abstract

A set of vertices W of a graph G is a total k-dominating set when every vertex of G has at least k neighbors in W. In a recent article, Chiarelli et al. (Improved Algorithms for k-Domination and Total k-Domination in Proper Interval Graphs, Lecture Notes in Comput. Sci. 10856, 290–302, 2018) prove that a total k-dominating set can be computed in $O(n^{3k})$ time when G is a proper interval graph with n vertices and m edges. In this note we reduce the time complexity to $O(m)$ for $k = 2$.

Keywords: total 2-domination, straight oriented graphs, proper interval graphs.

1 Introduction

A set of vertices W of a graph G is a *total k-dominating* set when every vertex v of G has at least k neighbors in W. The problem of computing a total k-dominating set of G with minimum cardinality is known to be NP-complete for every $k \geq 1$, even when G belongs to certain subclasses of chordal graphs [7] such as undirected path graphs [5, 6]. In turn, when G is an interval graph with n vertices, the problem is solvable in $O(n^{6k+4})$ time, as recently proven by Kang et al. [4] (cf. [2]). Moreover, the time complexity can be reduced to $O(n^{3k})$ when G belongs to the subclass of proper interval graphs [2].

Besides being a subclass of undirected path graphs, interval graphs are among the most famous classes of graphs. Unsurprisingly, then, the problem for $k = 1$ on interval graphs was studied long before the general case. In particular, Chang [1] shows that a total 1-dominating set of minimum cardinality can be obtained in $O(n)$ time when an interval model of G is given. The huge gap in the complexities of the algorithms by Chang, on the one hand, and Chiarelli et al. and Kang et al., on the other hand, suggests that there is still room for improvements when $k > 1$. One reason to explain this gap is the fact that the problems attacked by Kang et al. and Chiarelli et al. are too general. In this note we consider the problem from the opposite perspective, by studying the simplest case that is still unsolved. Specifically, we consider the total 2-domination problem on proper interval graphs, for which we obtain a quadratic ($O(m)$) time algorithm.

Our algorithm, as well as the one by Chiarelli et al. [2] and others, models the total 2-dominating problem as a shortest path problem on a weighted acyclic digraph D. The major difference is that, in the model by Chiarelli et al., each vertex of D represents a connected set of G with diameter at most 5. In turn, in our model each vertex represents different connected sets of varying diameters, that correspond to the weight of each outgoing edge and can be as high as $\Omega(n)$.

In Section 2 we introduce the terminology required throughout the paper. Then, in Section 3 we show how the total 2-domination problem is modeled as a shortest path problem on an acyclic digraph D of size $O(nm)$. We improve this model in Section 4, where we observe that D can be compressed to an acyclic digraph R, of size $O(m)$, that can be computed in $O(m)$ time. Finally, in Section 5 we discuss some ideas to try to generalize our algorithm to the case $k > 2$.

*CONICET and Departamento de Ciencia y Tecnología, Universidad Nacional de Quilmes, Bernal, Argentina.
†Supported by PICT ANPCyT grant 2015-2419.

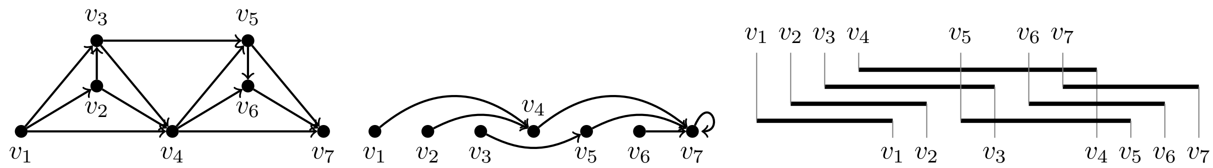


FIGURE 1. *A straight orientation of a PIG graph G (left) defined by an ordering $<$ and mapping f_r (center), and a corresponding family of inclusion-free intervals (right) that represent G.*

2 Preliminaries

In this article we work with simple graphs and digraphs. For a (di)graph G, let $V(G)$ and $E(G)$ denote the sets of vertices and (directed) edges of G, respectively. As usual, we write $n = |V(G)|$ and $m = |E(G)|$ when G is clear and, for simplicity, we use vw to denote both the set $\{v, w\}$ and the ordered pair (v, w). Two vertices v and w are *adjacent* when either $vw \in E(G)$ or $wv \in E(G)$. The *neighborhood* $N_G(v)$ of v is the set of all its adjacent vertices, while its *degree* is $d_G(v) = |N_G(v)|$. When G is a digraph, we say that $vw \in E(G)$ *goes from v to w*, while w is an *out-neighbor* of v. The *out-degree* of v is the number $d^+_G(v)$ of out-neighbors of v. For the sake of notation, we omit the subscript G from N and d when no confusions are possible.

A *path* in a (di)graph G is a sequence of vertices $P = v_1, \ldots, v_{k+1}$ such that $v_iv_{i+1} \in E(G)$, for $1 \le i \le k$. A *cycle* is a sequence $v_1, \ldots, v_{k+1}$ such that $v_1 = v_{k+1}$ and $v_1, \ldots, v_k$ is a path. If G has no cycles, then G is *acyclic*. We say that G is *weighted* to mean that each $e \in E(G)$ has a weight $\omega(e) \ge 0$. The *weight* of a path $P = v_1, \ldots, v_{k+1}$ is, then, $\omega(P) = \sum_{i=1}^{k} \omega(v_iv_{i+1})$. When G is a digraph, its *underlying graph* H has $V(G)$ as its vertex set, whereas $vw \in E(H)$ if and only if v and w are adjacent in G, for $v, w \in V(G)$. A graph G is *connected* when there is a path between every pair of vertices, while a digraph is *connected* when its underlying graph is connected. An *oriented graph* is a digraph G such that either $vw \notin E(G)$ or $wv \notin E(G)$, for $v, w \in V(G)$. If G is the underlying graph of an oriented graph $\vec{G}$, then $\vec{G}$ is an *orientation* of G.

Consider a (di)graph G. For $W \subseteq V(G)$, let $G[W]$ denote the sub(di)graph of G induced by W. We say that W is *connected* when $G[W]$ is connected. A *block* of W is a connected subset of W that is maximal by inclusion. We say that $v \in V(G)$ is 2*-dominated* by W when $|N(v) \cap W| \ge 2$. If every vertex of G is 2-dominated by W, then W is a 2*-dom*. Moreover, if $|W|$ is minimum among the 2-doms of G, then W is a *minimum* 2*-dom*. Note that W is a 2-dom of a graph G if and only it is a 2-dom of $\vec{G}$, for any orientation $\vec{G}$ of G. Thus, we may replace G by $\vec{G}$ when computing a minimum 2-dom. From now on we safely assume that $d(v) \ge 2$ ($v \in V(G)$); otherwise G has no 2-doms.

A *straight graph* (Figure 1) is an oriented graph G that admits a linear ordering $v_1 <_G \ldots <_G v_n$ of its vertices and a mapping $f^G_r \colon V(G) \to V(G)$ such that:

- $v_i \le_G f^G_r(v_i)$ for $1 \le i \le n$ and $f^G_r(v_i) \le_G f^G_r(v_{i+1})$ for $1 \le i < n$, and
- $v_iv_j \in E(G)$ if and only if $v_i <_G v_j \le_G f^G_r(v_i)$.

As before, G is usually omitted from $<$ and f_r. A graph G is a *proper interval* (PIG) *graph* is some of its orientations is a straight graph. It is well known that G is a PIG graph if and only if its vertices can be mapped into a family of inclusion-free intervals of the real line in such a way that two vertices of G are adjacent when their corresponding intervals have a nonempty intersection (e.g., [3]; see Figure 1). Yet, in this article we prefer the combinatorial view provided by straight graphs.

For the sake of notation, we sometimes assume that a straight graph G with vertices $v_1 < \ldots < v_n$ has two *artificial* vertices v_0 and v_{n+1}. Thus, for $0 \le i \le j \le n+1$, we can conveniently define the subsequence $G(v_i, v_j) = v_{i+1}, \ldots, v_{j-1}$. For $1 \le i \le n$, let $\ell^G(v_i) = v_{i-1}$, $r^G(v_i) = v_{i+1}$, and $u^G_r(v_i) = r^G(f_r(v_i))$, where the superscript G is omitted as usual. In colloquial terms, $\ell(v)$ and $r(v)$ are the vertices that precede and follow v, respectively, and $u_r(v)$ is the first vertex not adjacent to v in $G(v, v_{n+1})$. Also, define $f^0_r(v) = v$ and $f^{i+1}_r(v) = f_r(f^i_r(v))$ for $i > 0$. We extend $<$ from $V(G)$ to the family of subsets of $V(G)$ in such a way that, for $W_1, W_2 \subseteq (G)$, $W_1 < W_2$ if and only if $w_1 < w_2$ for every $w_1 \in W_1$ and $w_2 \in W_2$. It is not hard

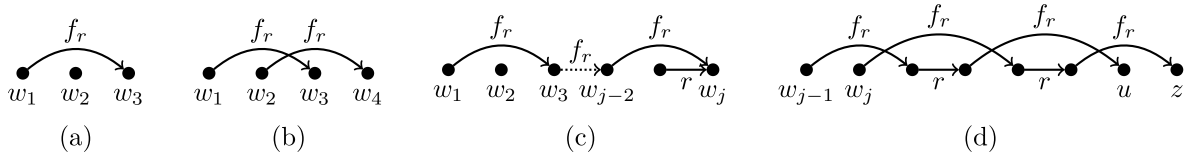


FIGURE 2. *The expansive connected set represented by w_1w_2 for sizes 3, 4, and j are depicted in (a), (b), and (c), respectively, whereas (d) describes the vertices u and z used to determine the j-extension vw of w_1w_2 for the case in which $u \neq z$.*

to see that the blocks of $W \subseteq V(G)$ are pairwise comparable by $<$. Therefore, we sometimes state that $B_1 < \ldots < B_k$ are the blocks of W.

3 Computing a minimum 2-dom in $O(nm)$ time

In this section we describe an algorithm to find a minimum 2-dom in $O(nm)$ time when a straight graph G is given. Let $v_1 < \ldots < v_n$ be the vertices of G, and define $S = \{v_1, v_2, v_3\}$ and $T = \{v_{n-2}, v_{n-1}, v_n\}$. To simplify the description of the algorithm, we assume that S and T are blocks of $V(G)$. Consequently, if W is a 2-dom of G with blocks $B_0 < \ldots < B_{k+1}$, then $B_0 = S$ and $B_{k+1} = T$. Note that this assumption yields no loss of generality because $B_1 \cup \ldots \cup B_k$ is a 2-dom of $G \setminus (B_0 \cup B_{k+1})$ if and only if $B_0 \ldots \cup B_{k+1}$ is a 2-dom of G. Thus, we can always transform the input graph by inserting $S \cup T$. The sets S and T are called the *source* and *pre-sink* blocks of G, respectively, whereas v_1v_2 and $v_{n-2}v_{n-1}$ are the *source* and *pre-sink* edges of G.

In a nutshell, the algorithm finds the blocks $B_1 < \ldots < B_k$ of the minimum 2-dom W one at a time, from B_1 to B_k. By the discussion above, B_1 is simply the source block of G. Once B_i is determined ($1 \leq i < k$), the next block B_{i+1} is obtained by choosing j vertices (for some $j \geq 3$) in a way that B_{i+1} reaches as far as possible. To build B_{i+1}, its first two vertices v and w are taken as the further reaching vertices that still cover the gap from B_i to B_{i+1}. Then, each of the remaining $j-2$ vertices are defined in terms of v and w. Under the terminology defined below, $B_1, \ldots, B_k$ and W are "expansive", while B_{i+1} "extends" B_i.

Formally, a connected set $B \subseteq V(G)$ with vertices $w_1 < \ldots < w_j$ ($j \geq 3$) is *expansive* (Figure 2) when:

(exp$_1$) $w_3 = f_r(w_1)$, $w_i = f_r^{i-2}(w_1)$ for $4 \leq i \leq j-2$, and $w_j = f_r(w_{j-2})$, and

(exp$_2$) if $j \geq 5$, then $w_{j-1} = \ell(w_j)$.

By (exp$_1$) and (exp$_2$), B is fully determined by w_1, w_2, and $|B|$; for this reason, we say that B is *represented* by w_1w_2. Clearly, w_1w_2 represents at most one expansive connected set of size j, $j \geq 0$. Let $u = f_r(u_r(w_{j-1}))$ and $z = f_r(u_r(w_j))$ (Figure 2). An expansive connected set B' represented by vw *extends* B when:

$$vw = \begin{cases} \ell(\ell(u))\ell(u) & \text{if } u = z \text{ and } d^+(\ell(u)) = 1 \\ \ell(u)u & \text{if } (u = z \text{ and } d^+(\ell(u)) > 1) \text{ or } (u \neq z \text{ and } d^+(u) = 1) \\ u\ell(z) & \text{if } u \neq z,\ f_r(u) = z \text{ and } d^+(u) > 1 \\ uz & \text{otherwise} \end{cases} \tag{1}$$

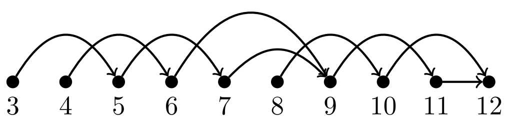
We refer to vw as being the *j-extension* of w_1w_2. Note that the j-extension of w_1w_2, if existing, is unique, because B is the unique expansive connected set with j vertices that is represented by w_1w_2. A set $W \subseteq V(G)$ with blocks $B_1 < \ldots < B_k$ is *expansive* when:

(exp$_3$) B_i is expansive for every $1 \leq i \leq k$, and

(exp$_4$) B_{i+1} extends B_i for every $1 \leq i < k$.

Moreover, if B_1 and B_k are the source and pre-sink blocks of G, then W is *fully expansive*.

Consider the weighted digraph D with vertex set $E(G)$ that has an edge from e to g of weight $\omega(eg) = j$



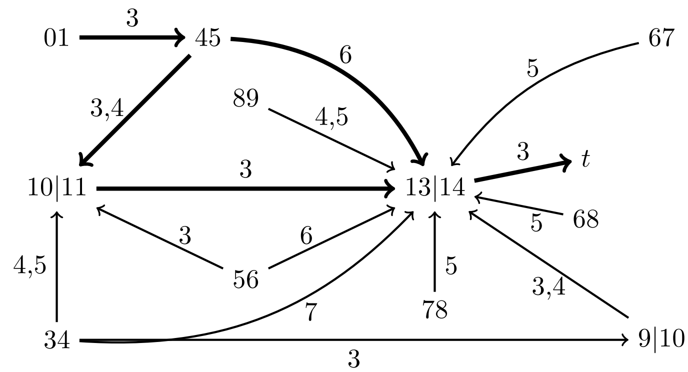


FIGURE 3. *Left: a straight graph G. Right: $D(G \cup S \cup T)$ for $S = \{0,1,2\}$ and $T = \{13,14,15\}$, where bold edges belong to paths of minimum weight from the source* 01 *to the sink t. The three fully expansive sets encoded by $D(G)$ are $S \cup T \cup \{4,5,6,10,11,12\}$, $S \cup T \cup \{4,5,6,7,10,11,12\}$, and $S \cup T \cup \{4,5,6,9,10,11\}$.*

when g is the j-extension of e.[1] Let e be the pre-sink of G and $t \notin E(G)$. Define $D(G)$ as the digraph that is obtained from D after the edge et with $\omega(et) = 3$ is inserted (Figure 3). The vertex t is the *sink* of $D(G)$, while its *source* is the source edge of G. By definition, $B < B'$ when B' extends B, thus $D(G)$ is acyclic. Moreover, any path $P = e_1, \ldots, e_{k+1}$ of $D(G)$ *encodes* an expansive set W with blocks $B_1 < \ldots < B_k$ such that B_i is represented by e_i and $|B_i| = \omega(e_ie_{i+1})$, for $1 \leq i \leq k$. By definition, $\omega(P) = \sum_{i=1}^{k} |B_i| = |W|$. We record the previous discussion for later.

Theorem 1. *If G is a straight graph, then $D(G)$ is an acyclic digraph that has $O(m)$ vertices and $O(nm)$ edges. Furthermore, $W \subseteq V(G)$ is (resp. fully) expansive if and only if W is encoded by a path of $D(G)$ (resp. from the source to the sink) whose weight is $|W|$.*

The key feature about fully expansive sets is that each of them is a 2-dom, while at least one of them is a minimum 2-dom. Of course, this claim holds only under our assumption that G has at least one 2-dom.

Theorem 2. *If $W \subseteq V(G)$ is a fully expansive set of a straight graph G, then W is a 2-dom.*

Proof. Suppose $W = w_1 < \ldots < w_k$ and let $w_0 < w_1$ and $w_{k+1} > w_k$ be the artificial vertices of G. Then, every vertex $v \in V(G)$ belongs to $G(w_i, w_j)$ for some $0 \leq i < j \leq k+1$. Take i and j so that $j - i$ is minimum, and consider the following cases for v.

Case 1: $i = 0$. This case is impossible, as w_1w_2 is the source edge of G and, consequently, $G(w_0, w_1) = \emptyset$.

Case 2: $j = k+1$. In this case, v is adjacent to w_{k-1} and w_k because $w_{k-1}w_k$ is the pre-sink edge of G.

Case 3: w_i and w_j belong to the same block of W. Then, either $v = w_{i+1}$ and $j = i+2$ or $j = i+1$. Whichever the case, v is adjacent to w_i and w_j.

Case 4: $v = w_{i+1}$ is the first of its block. Then, v is adjacent to w_{i+2} and $w_{i+3} = f_r(w_{i+1})$.

Case 5: $j = i+1$ and w_i and w_{i+1} belong to different blocks. By (1), $w_{i+1} \leq f_r(u_r(w_{i-1}))$ and $w_{i+2} \leq f_r(u_r(w_i))$. Thus, either $v < u_r(w_{i-1})$ is adjacent to w_{i-1} and w_i or $v \in G(f_r(w_{i-1}), u_r(w_i))$ is adjacent to w_i and w_{i+1} or $v \in G(f_r(w_i), w_{i+1})$ is adjacent to w_{i+1} and w_{i+2}.

As v is 2-dominated by W in every case, it follows that W is a 2-dom. □

Theorem 3. *If a straight graph G has a 2-dom, then it has a minimum 2-dom that is fully expansive.*

[1] As defined, D can contain multiple edges between the same pair of vertices (Figure 3). Moreover, some results hold only if D contains these repeated edges. Yet, for simplicity, we restricted our terminology to simple digraphs. This is not an issue, though, as all the results can be easily adapted to the case in which D is simple, by ignoring the heavier repeated edges. Is for this reason that we ignore the fact that D is a multidigraph.

Proof. Suppose $v_1 < \ldots < v_n$ are the vertices of G, and let $\pi(v_i) = i$, $1 \leq i \leq n$. For $W \subseteq V$, let $\pi(W) = \sum_{w \in W} \pi(w)$. We shall prove that every minimum 2-dom W with maximum π is fully expansive.

Consider any block B of W with vertices $w_1 < \ldots < w_j$ and let $u_3 = f_r(w_1)$, $u_i = f_r(w_{j-1})$ for $4 \leq i \leq j-2$, $u_j = f_r(w_{j-2})$, and $u_{j-1} = \ell(w_j)$ if $j \geq 5$. Following the same pattern as in Theorem 2, it is not hard to see that $W_i = (W \setminus \{w_i\}) \cup \{u_i\}$, $3 \leq i \leq j$, is a 2-dom of G. Moreover, since $|W| \leq |W_i|$ it follows that $u_i \notin W \setminus \{w_i\}$, hence $\pi(W_i) = \pi(W) + \pi(u_i) - \pi(w_i) \geq \pi(W)$. Consequently, $u_i = w_i$ by the maximality of $\pi(W)$. That is, B satisfies (exp$_1$) and (exp$_2$) and, thus, W satisfies (exp$_3$).

Suppose now that w_j is not the maximum of W. Then, some expansive block B' represented by an edge vw appears immediately after B in W. Let $u = f_r(u_r(w_{j-1}))$ and $z = f_r(u_r(w_j))$, and note that $u_r(w_j) \leq v$ because $B \cup B'$ is not connected. Since $u_r(w_{j-1})$ has at most one neighbor in B and $u_r(w_j)$ has no neighbors in B, it follows that (a) $v \leq u$ and (b) $w \leq z$. Moreover, (c) $f_r(v) \in B'$ by (exp$_1$). Suppose that (d) B' does not extend B, and consider the following cases.

Case 1: $u = z$ and $d^+(\ell(u)) = 1$. Since $d^+(v) \geq 2$, then $v \neq \ell(u)$. Then, by (a) and (b), it follows that $v \leq \ell(\ell(u))$. Moreover, as v has at least two neighbors in W, it follows that $w \leq \ell(u)$. Note that $W \setminus \{v, w\} \cup \{\ell(\ell(u)), \ell(u)\}$ is a 2-dom by (c) that, by (d) and (1), has $\pi > \pi(W)$.

Case 2: $u = z$ and $d^+(\ell(u)) > 1$. In this case, $v \leq \ell(u)$ by (a) and (b), while (c) implies that $W \setminus \{v, w, f_r(v)\} \cup \{\ell(u), u, f_r(\ell(u))\}$ is a 2-dom that, by (d) and (1), has $\pi > \pi(W)$.

Case 3: $u \neq z$ and $d^+(u) = 1$. Since $d^+(v) \geq 2$, then $v \neq u$. Then $v \leq \ell(u)$ by (a), while $w \leq u$ because v has at least two neighbors in W. Then, $(W \setminus \{v, w\}) \cup \{\ell(u), u\}$ is a 2-dom by (c) that has $\pi > \pi(W)$ by (d) and (1).

Case 4: $f_r(u) = z$ and $d^+(u) > 1$. Since v has at least two neighbors in W, (b) implies $w \leq \ell(z)$. Then, $(W \setminus \{v, w\}) \cup \{u, \ell(z)\}$ is a 2-dom by (c) that has $\pi > \pi(W)$ by (d) and (1).

Case 5: $u < z < f_r(u)$. In this final case, $W \setminus \{v, w, f_r(v)\} \cup \{u, z, f_r(u)\}$ is a 2-dom by (a)–(c) that, by (d) and (1), has $\pi > \pi(W)$.

As all the cases are impossible, B' extends B. Hence, (exp$_4$) holds as well. ☐

Theorems 1–3 imply that a minimum 2-dom can be obtained by computing a path of minimum weight from the source of $D(G)$ to its sink. By Theorem 1, this algorithm requires $O(nm)$ time once $D(G)$ is given. We remark that $D(G)$ can be generated in $O(nm)$ time, although the details are omitted as they are similar to those discussed in the next section for $R(G)$.

4 Computing a minimum 2-dom in $O(m)$ time

The idea to accelerate the algorithm is to compress $D(G)$ in a reduced graph $R(G)$ that uses two vertices per edge of G. For the sake of notation, let $\underline{\bar{E}}(G) = \underline{E}(G) \cup \bar{E}(G)$ for $\underline{E}(G) = \{\underline{e} \mid e \in E(G)\}$ and $\bar{E}(G) = \{\bar{e} \mid e \in E(G)\}$. Define the *width* of $vw \in E(G)$ as the minimum $\kappa \geq 1$ such that $d^+(f_r^\kappa(v)) \geq 2$; when no such κ exists, the width of vw is $\kappa = \infty$.

Let s and e be the source and pre-sink of G, respectively, and $t \notin E(G)$. As $D(G)$, the digraph $R(G)$ is obtained by inserting an edge $\underline{e}t$ of weight $\omega(\underline{e}t) = 3$ in a digraph R that, this time, has vertex set $\underline{\bar{E}}(G)$. The vertices $\underline{s}$ and t are the *source* and *sink* of $R(G)$. For each $e \in E(G)$ and $j \in \{3, 4\}$, R has *regular* edges $\underline{e}\underline{g}$ and $\underline{e}\bar{g}$ of weight j for each j-extension g of e. Similarly, if e has width $\kappa < \infty$, then R has *regular* edges $\bar{e}\underline{g}$ and $\bar{e}\bar{g}$ of weight $\kappa + 4$ when e has a $(\kappa+4)$-extension g. This time, however, the j-extensions of e for $j > \kappa + 4$ are compacted in a single edge. Specifically, if $e = vw$, $f_r(v) \neq w$, and $z = f_r^\kappa(v)$, then R has a *compact* edge $\bar{e}\bar{g}$ of weight κ for $g = zr(z)$. Figure 4 depicts $R(G)$ for the straight graph G in Figure 3.

The main feature of $R(G)$ is that it preserves the adjacencies and distances of $D(G)$. To make this assertion explicit, say that a path $P = \underline{\bar{e}}_1, \ldots, \underline{\bar{e}}_{h+1}$ of $R(G)$ is a D-*path* when $\underline{\bar{e}}_h\underline{\bar{e}}_{h+1}$ is regular, while it is a D-*edge* when it is a D-path and $\underline{\bar{e}}_i\underline{\bar{e}}_{i+1}$ is compact for $1 \leq i < h$. Clearly, any D-path P is equal to $P_1, \ldots, P_k$, where P_i is a D-edge from $\underline{\bar{e}}_i$ to $\underline{\bar{e}}_{i+1}$, for $1 \leq i \leq k$. By definition, $\underline{\bar{e}}_i \in \{\underline{e}_i, \bar{e}_i\}$ for some $e_i \in E(G)$. Following the terminology for $D(G)$, we say that P *encodes* the expansive set W with blocks

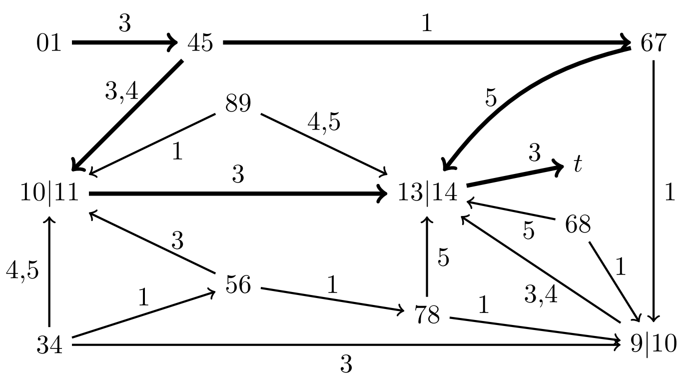


FIGURE 4. *$R(G \cup S \cup T)$ for G in Figure 3. For simplicity, $\underline{e}$ and $\bar{e}$ are depicted as one vertex e for every $e \in E(G)$. The edges of weight 3 and 4 are from $\underline{e}$ and those of weight 1 and 5 are from $\bar{e}$. Compact edges correspond to those of weight 1 and go to $\bar{e}$. Again, bold edges belong to paths of minimum weight from the source $\underline{01}$ to the sink t. The three fully expansive sets encoded by $R(G)$ are the same as those in Figure 3.*

$B_1 < \ldots < B_k$ such that B_i is represented by e_i and $|B_i| = \omega(P_i)$, for $1 \leq i \leq k$. Theorem 5 below is the translation of Theorem 1 to $R(G)$, that shows that $R(G)$ is actually a compact version of $D(G)$.

Theorem 4. *Let G be a straight graph, $e, g \in E(G)$, $\bar{\underline{g}} \in \{\underline{g}, \bar{g}\}$, and $j \in \mathbb{N}$. If $j < 5$, let $\bar{\underline{e}} = \underline{e}$; otherwise, let $\bar{\underline{e}} = \bar{e}$. Then, g is the j-extension of e if and only if there exists a D-edge from $\bar{\underline{e}}$ to $\bar{\underline{g}}$ of weight j in $R(G)$.*

Proof. Suppose first that g is the j-extension of e, and let κ be the width of e. We prove by induction on j that $R(G)$ has a D-edge of weight j from $\bar{\underline{e}}$ to $\bar{\underline{g}}$. The base case, in which $j \in \{3, 4, \kappa + 4\}$, is trivial as $\bar{\underline{e}}\bar{\underline{g}}$ is a regular edge of $R(G)$ with weight j. For the inductive step, let $B = w_1 < \ldots < w_j$ be the expansive connected set of size $j \geq 5$ that is represented by $e = w_1 w_2$. Note that B exists because otherwise e would have no j-extension. By (exp$_1$), $w_{j-2} = f_r^{j-4}(v)$, thus $\kappa < j - 4$ as $d^+(w_{j-2}) \geq 2$ and $j \neq \kappa + 4$. By definition, $R(G)$ has a compact edge from $\bar{\underline{e}}$ to $\bar{e}_\kappa$, for $e_\kappa = w_{\kappa+2} r(w_{\kappa+2})$, because $w_{\kappa+2} = f_r^\kappa(v)$ by (exp$_1$). Moreover, by (exp$_1$) and (exp$_2$), $B' = w_{\kappa+2}, r(w_{\kappa+2}), w_{\kappa+3}, \ldots, w_j$ is an expansive connected set with at least five vertices. By definition, g is the $(j - \kappa)$-extension of e_κ, thus, by induction, there is a D-edge P from $\bar{e}_\kappa$ to $\bar{\underline{g}}$ in $R(G)$ with $\omega(P) = j - \kappa$. Hence, $\bar{\underline{e}}P$ is a D-edge of $R(G)$ from $\bar{\underline{e}}$ to $\bar{\underline{g}}$ with $\omega(\bar{\underline{e}}P) = j$.

For the converse, suppose that $R(G)$ has a D-edge $P = \bar{\underline{e}}_0, \ldots, \bar{\underline{e}}_h$ from $\bar{\underline{e}}_0 = \bar{\underline{e}}$ to $\bar{\underline{e}}_h = \bar{\underline{g}}$ whose weight is j. Note that, by definition, $\bar{\underline{e}}_i = \bar{e}_i$ for every $1 < i < h$. We prove by induction on h that g is the j-extension of e. The base case $h = 1$ is trivial, because $\bar{\underline{e}}\bar{\underline{g}}$ is a regular edge of $R(G)$ only if g is the j-extension of e. For the inductive step, let κ be the width of $e = w_1 w_2$ and recall that $e_1 = w_{\kappa+2} r(w_{\kappa+2})$, where $w_{\kappa+2} = f_r^\kappa(w_1)$. By definition, $\bar{\underline{e}}_1, \ldots, \bar{\underline{e}}_h$ is a D-edge of $R(G)$ with weight $(j - \kappa)$, which implies that g is the $(j-\kappa)$-extension of e_1 by induction. Note that $j - \kappa \geq 5$ because $\bar{\underline{e}}_1 = \bar{e}_1$. Thus, by (exp$_1$) and (exp$_2$), e_1 represents an expansive connected set $B' = w_{\kappa+2} < r(w_{\kappa+2}) < w_{\kappa+3} < \ldots < w_j$ such that $w_i = f_r^{i-\kappa}(w_{\kappa+2}) = f_r^{i-2}(w_1)$ for every $\kappa \leq i \leq j - 2$, $w_j = f_r(w_{j-2})$, and $w_{j-1} = \ell(w_j)$. Therefore, by (exp$_1$) and (exp$_2$), $B = w_1, \ldots, w_j$ is an expansive connected set with $|B| = j$ when $w_i = f_r^{i-2}(w_1)$ for $3 \leq i \leq \kappa$. Consequently, by (1), e has g as its j-extension. □

Theorem 5. *If G is a straight graph, then $R(G)$ is an acyclic digraph that has $O(m)$ vertices and edges. Furthermore, $W \subseteq V(G)$ is (resp. fully) expansive if and only if W is encoded by a D-path of $R(G)$ (resp. from the source to the sink) whose weight is $|W|$.*

Proof. By Theorem 1, any expansive set W is encoded by a path $P = e_1, \ldots, e_{k+1}$ of $D(G)$. Let $h = k$ if e_{k+1} is the sink of $D(G)$ and $h = k+1$ otherwise. For $1 \leq i \leq h$, let $\bar{\underline{e}}_i = \underline{e}_i$ if $\omega(e_i e_{i+1}) < 5$ and $\bar{\underline{e}}_i = \bar{e}_i$ otherwise. By Theorem 4, there is a D-edge P_i from $\bar{\underline{e}}_i$ to $\bar{\underline{e}}_{i+1}$ of weight $\omega(e_i e_{i+1})$ for every $1 \leq i < h$. If $h = k$, then $\bar{\underline{e}}_k = \underline{e}_k$ because the unique edge form the pre-sink of G in $D(G)$ has weight 3. Thus, regardless of the value of h, the edge of $R(G)$ from $\bar{\underline{e}}_k$ to $\bar{\underline{e}}_{k+1}$ is a D-edge of weight $\omega(e_k e_{k+1})$. Consequently, $P_1, \ldots, P_k$ is a

D-path of $R(G)$ that encodes W. Moreover, if e_1 is the source edge of G, then $\omega(e_1e_2) = 3$ and, therefore, $\bar{e}_1 = \underline{e}_1$ is the source of $R(G)$ by Theorem 4. Hence, by Theorem 1, $P_1, \ldots, P_k$ goes from the source of $R(G)$ to its sink when W is fully expansive.

The converse is similar: if $P_1, \ldots, P_k$ is a D-path of $R(G)$ that encodes a set W, then $P = e_1, \ldots, e_{k+1}$ is a path of $D(G)$ by Theorem 4, where $e_i \in E(G)$ is the edge corresponding to the first edge of P_i for $1 \leq i \leq k$, and e_{k+1} corresponds to last edge of P_k that happens to be the sink of $D(G)$ when P_k ends at the sink of $R(G)$. Moreover, $\omega(P_i) = \omega(e_ie_{i+1})$. Thus, P encodes W which, by Theorem 1, implies that W is an expansive set of G with $\omega(P_1, \ldots, P_k)$ vertices. Moreover, W is fully expansive when $P_1, \ldots, P_k$ goes from the source to the sink of $R(G)$. ☐

The algorithm to compute a minimum 2-dom of a given straight graph G has three main steps. Steps 1 and 2 compute $R(G)$ and a path P of minimum weight from the source to the sink of $R(G)$, respectively. By Theorems 2–4, P encodes a minimum 2-dom W of G; the set W is found in Step 3. The algorithm runs in $O(m)$ time when implemented as described below, where we write $[n] = [1, n] \cap \mathbb{N}$.

Input: G is implemented with the sequence $v_1 < \ldots < v_n$ of its vertices and a function $\hat{f}_r \colon [n] \to [n]$ such that $\hat{f}_r(i) = j$ when $f_r(v_i) = v_j$. Both $V(G)$ and $\hat{f}_r$ are implemented with vectors, thus traversing $V(G)$ requires $O(n)$ time, whereas querying $\hat{f}_r(i)$ costs $O(1)$ time. Note that $\hat{d}^+(i) = d^+(v_i) = \hat{f}_r(i) - i$ can be answered in $O(1)$ time as well. We assume that G contains the source and pre-sink blocks, as $O(n)$ time suffices to insert them into the structure.

Step 0: before computing $R(G)$, we build the map $\hat{\kappa} \colon [n] \to [n+1] \times [n]$ such that: if v_i has width $\kappa < \infty$, then $\hat{\kappa}(i) = (\kappa, f_r^{\kappa}(v_i))$; otherwise, $\hat{\kappa}(i) = (n+1, i)$. A single backward traversal of $V(G)$ suffices to compute $\hat{\kappa}$ in $O(n)$ time because, by definition,

$$\hat{\kappa}(i) = \begin{cases} (n+1, i) & \text{if } \hat{d}^+(\hat{f}_r(i)) = 0 \\ (1, \hat{f}_r(i)) & \text{if } \hat{d}^+(\hat{f}_r(i)) \geq 2 \\ (1 + \min\{\kappa, n\}, w) & \text{otherwise, for } (\kappa, w) = \hat{\kappa}(\hat{f}_r(i)). \end{cases}$$

Step 1: to compute $R(G)$, first two vertices $(i, j, 0)$ and $(i, j, 1)$ representing $\underline{e}$ and $\bar{e}$ are created, for $e = v_iv_j \in E(G)$, $i \in [n]$, and $j \in (i, \hat{f}_r(i)]$. This step consumes $O(m)$ time. Then, for $i \in [n]$ and $j \in (i, f_r(i))$, the edges of $R(G)$ from $(i, j, 1)$ are inserted. Let $(\kappa, a) = \hat{\kappa}(i)$ and $b = \hat{f}_r(a)$; suppose $\kappa \leq n$ as no edge from $(i, j, 1)$ has to be inserted otherwise. First, the edge from $(i, j, 1)$ to $(a, a+1, 1)$, representing the compact edge $\bar{e}\bar{g}$ for $e = v_iv_j$ and $g = v_ar(v_a)$, is inserted in $O(1)$ time. To create the regular edges, note that, by (exp$_1$), v_{b-1} and v_b are the last two vertices of the expansive connected set of size $(\kappa + 4)$ that is represented by v_iv_j. Clearly, the indices x, y such that v_xv_y is the edge defined by (1), when applied to v_{b-1} and v_b, can be obtained in $O(1)$ time with a few applications of $\hat{f}_r$. By definition, v_xv_y is the $(\kappa + 4)$-extension of v_iv_j if and only if $v_bv_x \notin E(G)$ and v_xv_y represents an expansive connected set. These facts that can be determined in $O(1)$ time by observing whether $\hat{f}_r(b) < x$ and $\hat{d}^+(x) \geq 2$. If affirmative, then the edges from $(i, j, 1)$ to $(x, y, \bullet)$ of weight $\kappa + 4$ are inserted in $O(1)$ time. Therefore, the edges from $(i, j, 1)$ are created in $O(1)$ time. Each regular edge from $(i, j, 0)$ is inserted $O(1)$ time with a similar procedure. Therefore, this step consumes $O(m)$ total time.

Step 2: since $R(G)$ is acyclic (Theorem 4), P can be computed in $O(m)$ time.

Step 3: traversing P once, we can split P into D-edges $P_1, \ldots, P_k$ while $\omega(P_h)$ is computed for $h \in [k]$. Let $(i, j, \bullet)$ be the first vertex of P_h. By (exp$_1$) and (exp$_2$), $O(\omega(P_h))$ applications of $\hat{f}_r$ from i are enough to find all the vertices of the expansive connected set B_h of size $\omega(P_h)$ that is represented by v_iv_j. Then, the output $W = B_1 \cup \ldots \cup B_k$ can be computed in $O(\omega(P)) = O(n)$ time.

Note that G can be encoded in $O(n)$ space, thus the algorithm is quadratic in the worst case. We remark that many algorithms exist to compute a straight orientation $\vec{G}$ of a PIG graph G in $O(m)$ time. In particular, the algorithm in [3] outputs $\vec{G}$ as required by the algorithm above. Thus, when G is a PIG graph represented with adjacency lists, a 2-dom can be computed in linear time.

5 Concluding remarks

In this note we developed an $O(m)$ time algorithm for the total 2-dominating set problem on proper interval graphs, improving the previous $O(n^6)$ time algorithm by Chiarelli et al. [2]. Both of these algorithms work by finding a shortest path on a weigthed digraph D. The main difference between them is that in our model the edges of D represent connected sets with a large diameter. The actual connected set represented by $e \in D$ is the one that reaches farther in the input (model of the) graph G. One of the consequences defining the edges of D in this way is that some connected sets that can be a part of the solution when G is weighted are not considered. Therefore, on the contrary to the algorithm by Chiarelli et al., our algorithm does not solve the problem when G is weighted.

Our algorithm provides more evidence that the time required to solve problem of finding a total k-dominating set on a (proper) interval graph, for $k > 2$, is $o(n^{3k})$. In our digraph D, each edge goes from a pair vw to the another pair uz. Certainly, we can extend this model to k-tuples; the idea would be to have an edge from a k-tuple v to a k-tuple w of weight j when w is the further tuple that can be reached with a "block" having j vertices. Intuitively, such a k-tuple w should exist: if B and B' are two "blocks" of a total k-dominating set that begin with v and neither of them is lexicographically larger than the other, then it should be possible to combine B and B' into a new block beginning with v that is lexicographically larger than both B and B'. Thus, the tuple w reaching further should exist. The problem, however, is how to compute w when building D. The case $k = 2$ is easy because all the blocks have a peculiar structure. We conjecture that, by following these ideas, the problem can be solved in $O(n^k k^k)$ time.

References

[1] M.-S. Chang. Efficient algorithms for the domination problems on interval and circular-arc graphs. *SIAM J. Comput.*, 27(6):1671–1694, 1998. doi:10.1137/S0097539792238431.

[2] N. Chiarelli, T. R. Hartinger, V. A. Leoni, M. I. L. Pujato, and M. Milanic. Improved algorithms for k-domination and total k-domination in proper interval graphs. In J. Lee, G. Rinaldi, and A. R. Mahjoub, editors, *Combinatorial Optimization. ISCO 2018*, vol. 10856 of *Lecture Notes in Comput. Sci.*, pp. 290–302. Springer, Cham, 2018. doi:10.1007/978-3-319-96151-4_25.

[3] X. Deng, P. Hell, and J. Huang. Linear-time representation algorithms for proper circular-arc graphs and proper interval graphs. *SIAM J. Comput.*, 25(2):390–403, 1996. doi:10.1137/S0097539792269095.

[4] D. Y. Kang, O.-j. Kwon, T. J. F. Strø mme, and J. A. Telle. A width parameter useful for chordal and co-comparability graphs. *Theoret. Comput. Sci.*, 704:1–17, 2017. doi:10.1016/j.tcs.2017.09.006.

[5] J. K. Lan and G. J. Chang. On the algorithmic complexity of k-tuple total domination. *Discrete Appl. Math.*, 174:81–91, 2014. doi:10.1016/j.dam.2014.04.007.

[6] R. Laskar, J. Pfaff, S. M. Hedetniemi, and S. T. Hedetniemi. On the algorithmic complexity of total domination. *SIAM J. Algebraic Discrete Methods*, 5(3):420–425, 1984. doi:10.1137/0605040.

[7] D. Pradhan. Algorithmic aspects of k-tuple total domination in graphs. *Inform. Process. Lett.*, 112(21): 816–822, 2012. doi:10.1016/j.ipl.2012.07.010.