Improved Algorithms for k-Domination and Total k-Domination in Proper Interval Graphs

Nina Chiarelli[1,2], Tatiana Romina Hartinger[1,2], Valeria Alejandra Leoni[3,4(✉)], Maria Inés Lopez Pujato[3,5], and Martin Milanič[1,2(✉)]

[1] FAMNIT, University of Primorska, Glagoljaška 8, 6000 Koper, Slovenia
nina.chiarelli@famnit.upr.si, martin.milanic@upr.si
[2] IAM, University of Primorska, Muzejski trg 2, 6000 Koper, Slovenia
tatiana.hartinger@iam.upr.si
[3] FCEIA, Universidad Nacional de Rosario, Rosario, Santa Fe, Argentina
valeoni@fceia.unr.edu.ar, mineslpk@hotmail.com
[4] CONICET, Buenos Aires, Argentina
[5] ANPCyT, Buenos Aires, Argentina

Abstract. Given a positive integer k, a k-dominating set in a graph G is a set of vertices such that every vertex not in the set has at least k neighbors in the set. A total k-dominating set, also known as a k-tuple total dominating set, is a set of vertices such that every vertex of the graph has at least k neighbors in the set. The problems of finding the minimum size of a k-dominating, resp. total k-dominating set, in a given graph, are referred to as k-domination, resp. total k-domination. These generalizations of the classical domination and total domination problems are known to be NP-hard in the class of chordal graphs, and, more specifically, even in the classes of split graphs (both problems) and undirected path graphs (in the case of total k-domination). On the other hand, it follows from recent work by Kang et al. (2017) that these two families of problems are solvable in time $\mathcal{O}(|V(G)|^{6k+4})$ in the class of interval graphs. In this work, we develop faster algorithms for k-domination and total k-domination in the class of proper interval graphs. The algorithms run in time $\mathcal{O}(|V(G)|^{3k})$ for each fixed $k \geq 1$ and are also applicable to the weighted case.

Keywords: k-domination · Total k-domination
Proper interval graph · Polynomial-time algorithm

This work is supported in part by the Slovenian Research Agency (I0-0035, research program P1-0285 and research projects N1-0032, J1-6720, and J1-7051). The work for this paper was partly done in the framework of a bilateral project between Argentina and Slovenia, financed by the Slovenian Research Agency (BI-AR/15–17–009) and MINCYT-MHEST (SLO/14/09).

J. Lee et al. (Eds.): ISCO 2018, LNCS 10856, pp. 1–13, 2018.
https://doi.org/10.1007/978-3-319-96151-4_25

1 Introduction

Among the many variants of the domination problems [30,31], we consider in this paper a family of generalizations of the classical domination and total domination problems known as k-domination and total k-domination. Given a positive integer k and a graph G, a *k-dominating set* in G is a set $S \subseteq V(G)$ such that every vertex $v \in V(G) \setminus S$ has at least k neighbors in S, and a *total k-dominating set* in G is a set $S \subseteq V(G)$ such that every vertex $v \in V(G)$ has at least k neighbors in S. The k-domination and the total k-domination problems aim to find the minimum size of a k-dominating, resp. total k-dominating set, in a given graph. The notion of k-domination was introduced by Fink and Jacobson in 1985 [23] and studied in a series of papers (e.g., [14,20,22,27,42]) and in a survey by Chellali et al. [13]. The notion of total k-domination was introduced by Kulli in 1991 [41] and studied under the name of *k-tuple total domination* by Henning and Kazemi in 2010 [32] and also in a series of recent papers [1,39,43,53]. The terminology "k-tuple total domination" was introduced in analogy with the notion of "k-tuple domination", introduced in 2000 by Harary and Haynes [29].[1] The redundancy involved in k-domination and total k-domination problems can make them useful in various applications, for example in forming sets of representatives or in resource allocation in distributed computing systems (see, e.g., [31]). However, these problems are known to be NP-hard [37,53] and also hard to approximate [17].

The k-domination and total k-domination problems remain NP-hard in the class of chordal graphs. More specifically, the problems are NP-hard in the class of split graphs [42,53] and, in the case of total k-domination, also in the class of undirected path graphs [43]. We consider k-domination and total k-domination in another subclass of chordal graphs, the class of proper interval graphs. A graph G is an *interval graph* if it has an intersection model consisting of closed intervals on a real line, that is, if there exist a family $\mathcal{I}$ of intervals on the real line and a one-to-one correspondence between the vertices of G and the intervals of $\mathcal{I}$ such that two vertices are joined by an edge in G if and only if the corresponding intervals intersect. A *proper interval graph* is an interval graph that has a *proper interval model*, that is, an intersection model in which no interval contains another one. Proper interval graphs were introduced by Roberts [57], who showed that they coincide with the unit interval graphs, that is, graphs admitting an interval model in which all intervals are of unit length. Various characterizations of proper interval graphs have been developed in the literature (see, e.g., [24,26,36,49]) and several linear-time recognition algorithms are known, which in case of a yes instance also compute a proper interval model (see, e.g., [18] and references cited therein).

Domination and total domination problems are known to be solvable in linear time in the class of interval graphs (see [6,12,33] and [10,12,40,55,56], respectively). Furthermore, for each fixed integer $k \geq 1$, the k-domination and total

[1] A set S of vertices is said to be a *k-tuple dominating set* if every vertex of G is adjacent or equal to at least k vertices in S.

k-domination problems are solvable in time $\mathcal{O}(n^{6k+4})$ in the class of interval graphs where n is the order of the input graph. This follows from recent results due to Kang et al. [38], building on previous works by Bui-Xuan et al. [8] and Belmonte and Vatshelle [3]. In fact, Kang et al. studied a more general class of problems, called (ρ, σ)-domination problems, and showed that every such problem can be solved in time $\mathcal{O}(n^{6d+4})$ in the class of n-vertex interval graphs, where d is a parameter associated to the problem (see Corollary 3.2 in [38] and the paragraph following it).

1.1 Our Results and Approach

We significantly improve the above result for the case of proper interval graphs. We show that for each positive integer k, the k-domination and total k-domination problems are solvable in time $\mathcal{O}(n^{3k})$ in the class of n-vertex proper interval graphs. Except for $k = 1$, this improves on the best known running time.

Our approach is based on a reduction showing that for each positive integer k, the total k-domination problem on a given proper interval graph G can be reduced to a shortest path computation in a derived edge-weighted directed acyclic graph. A similar reduction works for k-domination. The reductions immediately result in algorithms with running time $\mathcal{O}(n^{4k+1})$. We show that with a suitable implementation the running time can be improved to $\mathcal{O}(n^{3k})$. The algorithms can be easily adapted to the weighted case, at no expense in the running time.

1.2 Related Work

We now give an overview of related work and compare our results with most relevant other results, besides those due to due to Kang et al. [38], which motivated this work.

Overview. For every positive integer k, the k-domination problem is NP-hard in the classes of bipartite graphs [2] and split graphs [42], but solvable in linear time in the class of graphs every block of which is a clique, a cycle or a complete bipartite graph (including trees, block graphs, cacti, and block-cactus graphs) [42], and, more generally, in any class of graphs of bounded clique-width [19,50] (see also [16]). The total k-domination problem is NP-hard in the classes of split graphs [53], doubly chordal graphs [53], bipartite graphs [53], undirected path graphs [43], and bipartite planar graphs (for $k \in \{2, 3\}$) [1], and solvable in linear time in the class of graphs every block of which is a clique, a cycle, or a complete bipartite graph [43], and, more generally, in any class of graphs of bounded clique-width [19,50], and in polynomial time in the class of chordal bipartite graphs [53]. k-domination and total k-domination problems were also studied with respect to their (in)approximability properties, both in general [17] and in restricted graph classes [2], as well as from the parameterized complexity point of view [9,34].

Besides k-domination and total k-domination, other variants of domination problems solvable in polynomial time in the class of proper interval graphs (or in some of its superclasses) include k-tuple domination for all $k \geq 1$ [45] (see also [44] and, for $k = 2$, [54]), connected domination [56], independent domination [21], paired domination [15], efficient domination [11], liar's domination [51], restrained domination [52], eternal domination [5], power domination [46], outer-connected domination [48], Roman domination [47], Grundy domination [7], etc.

Comparison. Bertossi [4] showed how to reduce the total domination problem in a given interval graph to a shortest path computation in a derived edge-weighted directed acyclic graph satisfying some additional constraints on pairs of consecutive arcs. A further transformation reduces the problem to a usual (unconstrained) shortest path computation. Compared to the approach of Bertossi, our approach exploits the additional structure of proper interval graphs in order to gain generality in the problem space. Our approach works for every k and is also more direct, in the sense that the (usual or total, unweighted or weighted) k-domination problem in a given proper interval graph is reduced to a shortest path computation in a derived edge-weighted acyclic digraph in a single step.

The works of Liao and Chang [45] and of Lee and Chang [44] consider various domination problems in the class of strongly chordal graphs (and, in the case of [45], also dually chordal graphs). While the class of strongly chordal graphs generalizes the class of interval graphs, the domination problems studied in [44, 45] all deal with closed neighborhoods, and for those cases structural properties of strongly chordal and dually chordal graphs are helpful for the design of linear-time algorithms. In contrast, k-domination and total k-domination are defined via open neighborhoods and results of [44,45] do not seem to be applicable or easily adaptable to our setting.

Structure of the Paper. In Sect. 2, we describe the reduction for the total k-domination problem. The specifics of the implementation resulting in improved running time are given in Sect. 3. In Sect. 4, we discuss how the approach can be modified to solve the k-domination problem and the weighted cases. We conclude the paper with some open problems in Sect. 5. Due to lack of space, most proofs are omitted. They can be found in the full version [58].

In the rest of the section, we fix some definitions and notation. Given a graph G and a set $X \subseteq V(G)$, we denote by $G[X]$ the subgraph of G induced by X and by $G - X$ the subgraph induced by $V(G) \setminus X$. For a vertex u in a graph G, we denote by $N(u)$ the set of neighbors of u in G. Note that for every graph G, the set $V(G)$ is a k-dominating set, while G has a total k-dominating set if and only if every vertex of G has at least k neighbors.

2 The Reduction for Total k-Domination

Let k be a positive integer and $G = (V, E)$ be a given proper interval graph. We will assume that G is equipped with a proper interval model $\mathcal{I} = \{I_j \mid j = 1, \ldots, n\}$ where $I_j = [a_j, b_j]$ for all $j = 1, \ldots, n$. (As mentioned in the introduction, a proper interval model of a given proper interval graph can be computed in linear time.) We may also assume that no two intervals coincide. Moreover, since in a proper interval model the order of the left endpoints equals the order of the right endpoints, we assume that the intervals are sorted increasingly according to their left endpoints, i.e., $a_1 < \ldots < a_n$. We use notation $I_j < I_\ell$ if $j < \ell$ and say in this case that I_j is *to the left of* I_ℓ and I_ℓ is *to the right of* I_j. Also, we write $I_j \leq I_\ell$ if $j \leq \ell$. Given three intervals $I_j, I_\ell, I_m \in \mathcal{I}$, we say that interval I_ℓ is *between* intervals I_j and I_m if $j < \ell < m$. We say that interval I_j *intersects* interval I_ℓ if $I_j \cap I_\ell \neq \emptyset$.

Our approach can be described as follows. Given G, we compute an edge-weighted directed acyclic graph D_k^t (where the superscript "t" means "total") and show that the total k-domination problem on G can be reduced to a shortest path computation in D_k^t. The definition of the digraph given next is followed by an example and an explanation of the intuition behind the reduction.

To distinguish the vertices of D_k^t from those of G, we will refer to them as *nodes*. Vertices of G will be typically denoted by u or v, and nodes of D_k^t by s, s', s''. Each node of D_k^t will be a sequence of intervals from the set $\mathcal{I}' = \mathcal{I} \cup \{I_0, I_{n+1}\}$, where I_0, I_{n+1} are two new, "dummy" intervals such that $I_0 < I_1$, $I_0 \cap I_1 = \emptyset$, $I_n < I_{n+1}$, and $I_n \cap I_{n+1} = \emptyset$. We naturally extend the linear order $<$ on $\mathcal{I}$ to the whole set $\mathcal{I}'$. We will say that an interval $I \in \mathcal{I}'$ is *associated with* a node s of D_k^t if it appears in sequence s. The set of all intervals associated with s will be denoted by $\mathcal{I}_s$. Given a node s of D_k^t, we will denote by $\min(s)$ and $\max(s)$ the first, resp., the last interval in $\mathcal{I}_s$ with respect to ordering $<$ of $\mathcal{I}'$. A sequence $(I_{i_1}, \ldots, I_{i_q})$ of intervals from $\mathcal{I}$ is said to be *increasing* if $i_1 < \ldots < i_q$.

The node set of D_k^t is given by $V(D_k^t) = \{I_0, I_{n+1}\} \cup S \cup B$, where:

- I_0 and I_{n+1} are sequences of intervals of length one.[2]
- S is the set of so-called *small nodes*. Set S consists exactly of those increasing sequences $s = (I_{i_1}, \ldots, I_{i_q})$ of intervals from $\mathcal{I}$ such that:
  (1) $k + 1 \leq q \leq 2k - 1$,
  (2) for all $j \in \{1, \ldots, q-1\}$, we have $I_{i_j} \cap I_{i_{j+1}} \neq \emptyset$, and
  (3) every interval $I \in \mathcal{I}$ such that $\min(s) \leq I \leq \max(s)$ intersects at least k intervals from the set $\mathcal{I}_s \setminus \{I\}$.
- B is the set of so-called *big nodes*. Set B consists exactly of those increasing sequences $s = (I_{i_1}, \ldots, I_{i_{2k}})$ of intervals from $\mathcal{I}$ of length $2k$ such that:
  (1) for all $j \in \{1, \ldots, 2k-1\}$, we have $I_{i_j} \cap I_{i_{j+1}} \neq \emptyset$, and
  (2) every interval $I \in \mathcal{I}$ such that $I_{i_k} \leq I \leq I_{i_{k+1}}$ intersects at least k intervals from the set $\mathcal{I}_s \setminus \{I\}$.

The arc set of D_k^t is given by $E(D_k^t) = E_0 \cup E_1$, where:

- Set E_0 consists exactly of those ordered pairs $(s, s') \in V(D_k^t) \times V(D_k^t)$ such that:
  (1) $\max(s) < \min(s')$ and $\max(s) \cap \min(s') = \emptyset$,

[2] This assures that the intervals $\min(s)$ and $\max(s)$ are well defined also for $s \in \{I_0, I_{n+1}\}$, in which case both are equal to s.

(2) every interval $I \in \mathcal{I}$ such that $\max(s) < I < \min(s')$ intersects at least k intervals from $\mathcal{I}_s \cup \mathcal{I}_{s'}$,
(3) if $s \in B$, then the rightmost $k+1$ intervals associated with s pairwise intersect, and
(4) if $s' \in B$, then the leftmost $k+1$ intervals associated with s' pairwise intersect.

– Set E_1 consists exactly of those ordered pairs $(s, s') \in V(D_k^t) \times V(D_k^t)$ such that $s, s' \in B$ and there exist $2k+1$ intervals $I_{i_1}, \ldots, I_{i_{2k+1}}$ in $\mathcal{I}$ such that $s = (I_{i_1}, I_{i_2}, \ldots, I_{i_{2k}})$ and $s' = (I_{i_2}, I_{i_3}, \ldots, I_{i_{2k+1}})$.

To every arc (s, s') of D_k^t we associate a non-negative length $\ell(s, s')$, defined as follows:

$$\ell(s, s') = \begin{cases} |\mathcal{I}_{s'}|, & \text{if } (s, s') \in E_0 \text{ and } s' \neq I_{n+1}; \\ 1, & \text{if } (s, s') \in E_1; \\ 0, & \text{otherwise.} \end{cases} \qquad (*)$$

The length of a directed path in D_k^t is defined, as usual, as the sum of the lengths of its arcs.

Example 1. Consider the problem of finding a minimum total 2-dominating set in the graph G given by the proper interval model $\mathcal{I}$ depicted in Fig. 1(a). Using the reduction described above, we obtain the digraph D_2^t depicted in Fig. 1(c), where, for clarity, nodes $(I_{i_1}, \ldots, I_{i_p})$ of D_2^t are identified with the corresponding strings of indices $i_1 i_2 \ldots i_p$. We also omit in the figure the (irrelevant) nodes that do not belong to any directed path from I_0 to I_{n+1}. There is a unique shortest I_0, I_9-path in D_2^t, namely $(0, 2356, 3567, 9)$. The path corresponds to $\{2, 3, 5, 6, 7\}$, the only minimum total 2-dominating set in G.

The correctness of the above reduction is established by proving the following.

Proposition 1. *Given a proper interval graph G and a positive integer k, let D_k^t be the directed graph constructed as above. Then G has a total k-dominating set of size c if and only if D_k^t has a directed path from I_0 to I_{n+1} of length c.*

The intuition behind the reduction is the following. The subgraph of G induced by a minimum total k-dominating set splits into connected components. These components as well as vertices within them are naturally ordered from left to right. Moreover, since each connected subgraph of a proper interval graph has a Hamiltonian path, the nodes of D_k^t correspond to paths in G, see condition (2) for small nodes or condition (1) for big nodes. Since each vertex of G has at least k neighbors in the total k-dominating set, each component has at least $k+1$ vertices. Components with at least $2k$ vertices give rise to directed paths in D_k^t consisting of big nodes and arcs in E_1. Each component with less than $2k$ vertices corresponds to a unique small node in D_k^t, which can be seen as a trivial directed path in D_k^t. The resulting paths inherit the left-to-right ordering from the components and any two consecutive paths are joined in D_k^t by an arc in E_0. Moreover, I_0 is joined to the leftmost node of the leftmost path with an arc in E_0 and, symmetrically, the rightmost node of the rightmost path is joined to

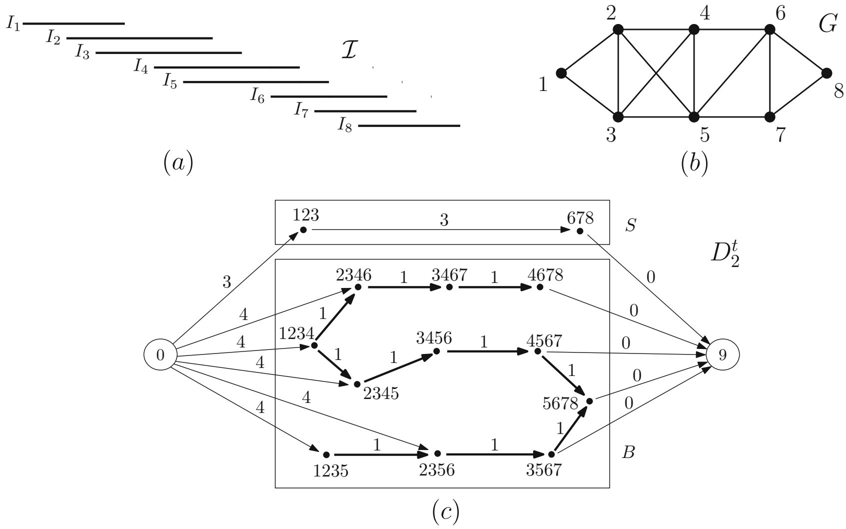

Fig. 1. (a) A proper interval model $\mathcal{I}$, (b) the corresponding proper interval graph G, and (c) a part of the derived digraph D_2^t, where only nodes that lie on some directed path from I_0 to I_9 are shown. Edges in E_1 are depicted bold.

I_{n+1} with an arc in E_0. Adding such arcs yields a directed path from I_0 to I_{n+1} of the desired length.

The above process can be reversed. Given a directed path P in D_k^t from I_0 to I_{n+1}, a total k-dominating set in G of the desired size can be obtained as the set of all vertices corresponding to intervals in $\mathcal{I}$ associated with internal nodes of P. The total k-dominating property is established using the defining properties of small nodes, big nodes, and arcs in E_0 and in E_1. For example, condition (3) in the definition of arcs in E_0 guarantees that the vertex corresponding to the rightmost interval associated with $s \in B$ where $(s, s') \in E_0$ is k-dominated. The condition is related to the fact that in proper interval graphs the neighborhood of a vertex represented by an interval $[a, b]$ splits into two cliques: one for all intervals containing a and another one for all intervals containing b.

The digraph D_k^t has $\mathcal{O}(n^{2k})$ nodes and $\mathcal{O}(n^{4k})$ arcs and can be, together with the length function ℓ on its arcs, computed from G directly from the definition in time $\mathcal{O}(n^{4k+1})$. A shortest directed path (with respect to ℓ) from I_0 to all nodes reachable from I_0 in D_k^t can be computed in polynomial time using any of the standard approaches, for example using Dijkstra's algorithm. Actually, since D_k^t is acyclic, a dynamic programming approach along a topological ordering of D_k^t can be used to compute shortest paths from I_0 in linear time (in the size of D_k^t). Proposition 1 therefore implies that the total k-domination problem is solvable in time $\mathcal{O}(n^{4k+1})$ in the class of n-vertex proper interval graphs.

We will show in the next section that, with a careful implementation, a shortest I_0, I_{n+1}-path in D_k^t can be computed without examining all the arcs of the digraph, leading to the claimed improvement in the running time.

3 Improving the Running Time

We assume all notations from Sect. 2. In particular, G is a given n-vertex proper interval graph equipped with a proper interval model $\mathcal{I}$ and (D_k^t, ℓ) is the derived edge-weighted acyclic digraph with $\mathcal{O}(n^{2k})$ nodes. We apply Proposition 1 and show that a shortest I_0, I_{n+1}-path in D_k^t can be computed in time $\mathcal{O}(n^{3k})$. The main idea of the speedup relies on the fact that the algorithm avoids examining all arcs of the digraph. This is achieved by employing a dynamic programming approach based on a partition of a subset of the node set into $\mathcal{O}(n^k)$ parts depending on the nodes' suffixes of length k. The partition will enable us to efficiently compute minimum lengths of four types of directed paths in D_k^t, all starting in I_0 and ending in a specified vertex, vertex set, arc, or arc set. In particular, a shortest I_0, I_{n+1}-path in D_k^t will be also computed this way.

Theorem 1. *For every positive integer k, the total k-domination problem is solvable in time $\mathcal{O}(|V(G)|^{3k})$ in the class of proper interval graphs.*

Proof (sketch). By Proposition 1, it suffices to show that a shortest directed path from I_0 to I_{n+1} in D_k^t can be computed in the stated time. Due to lack of space, we only explain some implementation details. In order to describe the algorithm, we need to introduce some notation. Given a node $s \in S \cup B$, say $s = (I_{i_1}, \ldots, I_{i_q})$ (recall that $k+1 \leq q \leq 2k$), we define its *k-suffix* of s as the sequence $(I_{i_{q-k+1}}, \ldots, I_{i_q})$ and denote it by $\text{suf}_k(s)$.

The algorithm proceeds as follows. First, it computes the node set of D_k^t and a subset B' of the set of big nodes consisting of precisely those nodes $s \in B$ satisfying condition (3) in the definition of E_0 (that is, the rightmost $k+1$ intervals associated with s pairwise intersect). Next, it computes a partition $\{A_\sigma \mid \sigma \in \Sigma\}$ of $S \cup B'$ defined by $\Sigma = \{\text{suf}_k(s) : s \in S \cup B'\}$ and $A_\sigma = \{s \in S \cup B' \mid \text{suf}_k(s) = \sigma\}$ for all $\sigma \in \Sigma$.

The algorithm also computes the arc set E_1. On the other hand, the arc set E_0 is not generated explicitly, except for the arcs in E_0 with tail I_0 or head I_{n+1}. Using dynamic programming, the algorithm will compute the following values.

(i) For all $s \in V(D_k^t) \setminus \{I_0\}$, let p_s^0 denote the minimum ℓ-length of a directed I_0,s-path in D_k^t ending with an arc from E_0.
(ii) For all $s \in V(D_k^t) \setminus \{I_0\}$, let p_s denote the minimum ℓ-length of a directed I_0,s-path in D_k^t.
(iii) For all $e \in E_1$, let p_e denote the minimum ℓ-length of a directed path in D_k^t starting in I_0 and ending with e.
(iv) For all $\sigma \in \Sigma$, let p_σ denote the minimum ℓ-length of a directed path in D_k^t starting in I_0 and ending in A_σ.

In all cases, if no path of the corresponding type exists, we set the value of the respective p_s^0, p_s, p_e, or p_σ to ∞.

Clearly, once all the p_s^0, p_s, p_e, and p_σ values will be computed, the length of a shortest I_0, I_{n+1}-path in D_k^t will be given by $p_{I_{n+1}}$.

The above values can be computed using the following recursive formulas:

(i) p_s^0 values:
- For $s \in S \cup B$, let $\Sigma_s = \{\sigma \in \Sigma \mid (\tilde{s}, s) \in E_0 \text{ for some } \tilde{s} \in A_\sigma\}$ and set

$$p_s^0 = \begin{cases} |\mathcal{I}_s|, & \text{if } (I_0, s) \in E_0; \\ \min\limits_{\sigma \in \Sigma_s} p_\sigma + |\mathcal{I}_s|, & \text{if } (I_0, s) \notin E_0 \text{ and } \Sigma_s \neq \emptyset; \\ \infty, & \text{otherwise.} \end{cases}$$

- For $s = I_{n+1}$, let $p_s^0 = \min\limits_{(\tilde{s},s) \in E_0} p_{\tilde{s}}$.

(ii) p_s values: For all $s \in V(D_k^t) \setminus \{I_0\}$, we have $p_s = \min\left\{p_s^0, \min\limits_{(\tilde{s},s) \in E_1} p_{(\tilde{s},s)}\right\}$.

(iii) p_e values: For all $e = (s, s') \in E_1$, we have $p_e = p_s + 1$.

(iv) p_σ values: For all $\sigma \in \Sigma$, we have $p_\sigma = \min\limits_{s \in A_\sigma} p_s$.

The above formulas can be computed following any topological sort of D_k^t such that if $s, s' \in S \cup B$ are such that $\mathrm{suf}_k(s) \neq \mathrm{suf}_k(s')$ and $\mathrm{suf}_k(s)$ is lexicographically smaller than $\mathrm{suf}_k(s')$, then s appears strictly before s' in the ordering. When the algorithm processes a node $s \in V(D_k^t) \setminus \{I_0\}$, it computes the values of p_s^0, p_e for all $e = (\tilde{s}, s) \in E_1$, and p_s, in this order. For every $\sigma \in \Sigma$, the value of p_σ is computed as soon as the values of p_s are known for all $s \in A_\sigma$. This completes the description of the algorithm. □

4 Modifying the Approach for *k*-Domination and for Weighted Problems

With minor modifications of the definitions of small nodes, big nodes, and arcs in E_0 of the derived digraph, the approach developed in Sects. 2 and 3 for total k-domination leads to an analogous result for k-domination.

Theorem 2. *For every positive integer k, the k-domination problem is solvable in time $\mathcal{O}(|V(G)|^{3k})$ in the class of proper interval graphs.*

The approach of Kang et al. [38], which implies that k-domination and total k-domination are solvable in time $O(|V(G)|^{6k+4})$ in the class of interval graphs also works for the weighted versions of the problems, where each vertex $u \in V(G)$ is equipped with a non-negative cost $c(u)$ and the task is to find a (usual or total) k-dominating set of G of minimum total cost. For both families of problems, our approach can also be easily adapted to the weighed case. Denoting the total cost of a set $\mathcal{J}$ of vertices (i.e., intervals) by $c(\mathcal{J}) = \sum_{I\in\mathcal{J}} c(I)$, it suffices to generalize the length function from (∗) in a straightforward way, as follows:

$$\ell(s,s') = \begin{cases} c(\mathcal{I}_{s'}), & \text{if } (s,s') \in E_0 \text{ and } s' \neq I_{n+1}; \\ c(\min(s')), & \text{if } (s,s') \in E_1; \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

This results in $O(|V(G)|^{3k})$ algorithms for the weighted (usual or total) k-domination problems in the class of proper interval graphs.

5 Conclusion

In this work, we presented improved algorithms for weighted k-domination and total k-domination problems for the class of proper interval graphs. The time complexity was significantly improved, from $\mathcal{O}(n^{6k+4})$ to $\mathcal{O}(n^{3k})$, for each fixed integer $k \geq 1$. Our work leaves open several questions. Even though polynomial for each fixed k, our algorithms are too slow to be of practical use, and the main question is whether having k in the exponent of the running time can be avoided. Are the k-domination and total k-domination problems fixed-parameter tractable with respect to k in the class of proper interval graphs? Could it be that even the more general problems of *vector domination* and *total vector domination* (see, e.g., [17,25,28,35]), which generalize k-domination and total k-domination when k is part of input, can be solved in polynomial time in the class of proper interval graphs? It would also be interesting to determine the complexity of these problems in generalizations of proper interval graphs such as interval graphs, strongly chordal graphs, cocomparability graphs, and AT-free graphs.

References

1. Argiroffo, G., Leoni, V., Torres, P.: Complexity of k-tuple total and total $\{k\}$-dominations for some subclasses of bipartite graphs Inform. Process. Lett. (2017). https://doi.org/10.1016/j.ipl.2018.06.007
2. Bakhshesh, D., Farshi, M., Hasheminezhad, M.: Complexity results for k-domination and α-domination problems and their variants (2017). arXiv:1702.00533 [cs.CC]
3. Belmonte, R., Vatshelle, M.: Graph classes with structured neighborhoods and algorithmic applications. Theoret. Comput. Sci. **511**, 54–65 (2013)
4. Bertossi, A.A.: Total domination in interval graphs. Inform. Process. Lett. **23**(3), 131–134 (1986)
5. Braga, A., de Souza, C.C., Lee, O.: The eternal dominating set problem for proper interval graphs. Inform. Process. Lett. **115**(6–8), 582–587 (2015)
6. Brandstädt, A., Chepoi, V.D., Dragan, F.F.: The algorithmic use of hypertree structure and maximum neighbourhood orderings. Discrete Appl. Math. **82**(1–3), 43–77 (1998)

7. Brešar, B., Gologranc, T., Kos, T.: Dominating sequences under atomic changes with applications in sierpiński and interval graphs. Appl. Anal. Discrete Math. **10**(2), 518–531 (2016)
8. Bui-Xuan, B.-M., Telle, J.A., Vatshelle, M.: Fast dynamic programming for locally checkable vertex subset and vertex partitioning problems. Theoret. Comput. Sci. **511**, 66–76 (2013)
9. Cattanéo, D., Perdrix, S.: The parameterized complexity of domination-type problems and application to linear codes. In: Gopal, T.V., Agrawal, M., Li, A., Cooper, S.B. (eds.) TAMC 2014. LNCS, vol. 8402, pp. 86–103. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-06089-7_7
10. Chang, G.J.: Labeling algorithms for domination problems in sun-free chordal graphs. Discrete Appl. Math. **22**(1), 21–34 (1989)
11. Chang, G.J., Pandu, C.P., Coorg, S.R.: Weighted independent perfect domination on cocomparability graphs. Discrete Appl. Math. **63**(3), 215–222 (1995)
12. Chang, M.-S.: Efficient algorithms for the domination problems on interval and circular-arc graphs. SIAM J. Comput. **27**(6), 1671–1694 (1998)
13. Chellali, M., Favaron, O., Hansberg, A., Volkmann, L.: k-domination and k-independence in graphs: a survey. Graphs Combin. **28**(1), 1–55 (2012)
14. Chellali, M., Meddah, N.: Trees with equal 2-domination and 2-independence numbers. Discuss. Math. Graph Theory **32**(2), 263–270 (2012)
15. Cheng, T.C.E., Kang, L.Y., Ng, C.T.: Paired domination on interval and circular-arc graphs. Discrete Appl. Math. **155**(16), 2077–2086 (2007)
16. Cicalese, F., Cordasco, G., Gargano, L., Milanič, M., Vaccaro, U.: Latency-bounded target set selection in social networks. Theoret. Comput. Sci. **535**, 1–15 (2014)
17. Cicalese, F., Milanič, M., Vaccaro, U.: On the approximability and exact algorithms for vector domination and related problems in graphs. Discrete Appl. Math. **161**(6), 750–767 (2013)
18. Corneil, D.G.: A simple 3-sweep LBFS algorithm for the recognition of unit interval graphs. Discrete Appl. Math. **138**(3), 371–379 (2004)
19. Courcelle, B., Makowsky, J.A., Rotics, U.: Linear time solvable optimization problems on graphs of bounded clique-width. Theory Comput. Syst. **33**(2), 125–150 (2000)
20. DeLaViña, E., Goddard, W., Henning, M.A., Pepper, R., Vaughan, E.R.: Bounds on the k-domination number of a graph. Appl. Math. Lett. **24**(6), 996–998 (2011)
21. Farber, M.: Independent domination in chordal graphs. Oper. Res. Lett. **1**(4), 134–138 (1982)
22. Favaron, O., Hansberg, A., Volkmann, L.: On k-domination and minimum degree in graphs. J. Graph Theory **57**(1), 33–40 (2008)
23. Fink, J.F., Jacobson, M.S.: n-domination in graphs. In: Graph Theory with Applications to Algorithms and Computer Science (Kalamazoo, Mich., 1984), pp. 283–300. Wiley, New York (1985)
24. Gardi, F.: The roberts characterization of proper and unit interval graphs. Discrete Math. **307**(22), 2906–2908 (2007)
25. Gerlach, T., Harant, J.: A note on domination in bipartite graphs. Discuss. Math. Graph Theory **22**(2), 229–231 (2002)
26. Gutierrez, M., Oubiña, L.: Metric characterizations of proper interval graphs and tree-clique graphs. J. Graph Theory **21**(2), 199–205 (1996)
27. Hansberg, A., Pepper, R.: On k-domination and j-independence in graphs. Discrete Appl. Math. **161**(10–11), 1472–1480 (2013)
28. Harant, J., Pruchnewski, A., Voigt, M.: On dominating sets and independent sets of graphs. Combin. Probab. Comput. **8**(6), 547–553 (1999)

29. Harary, F., Haynes, T.W.: Double domination in graphs. Ars Combin. **55**, 201–213 (2000)
30. Haynes, T.W., Hedetniemi, S.T., Slater, P.J. (eds.): Domination in Graphs. Advanced Topics, volume 209 of Monographs and Textbooks in Pure and Applied Mathematics. Marcel Dekker Inc., New York (1998)
31. Haynes, T.W., Hedetniemi, S.T., Slater, P.J.: Fundamentals of domination in graphs. In: Monographs and Textbooks in Pure and Applied Mathematics, vol. 208. Marcel Dekker Inc., New York (1998)
32. Henning, M.A., Kazemi, A.P.: k-tuple total domination in graphs. Discrete Appl. Math. **158**(9), 1006–1011 (2010)
33. Hsu, W.L., Tsai, K.-H.: Linear time algorithms on circular-arc graphs. Inform. Process. Lett. **40**(3), 123–129 (1991)
34. Ishii, T., Ono, H., Uno, Y.: Subexponential fixed-parameter algorithms for partial vector domination. Discrete Optim. **22**(part A), 111–121 (2016)
35. Ishii, T., Ono, H., Uno, Y.: (Total) vector domination for graphs with bounded branchwidth. Discrete Appl. Math. **207**, 80–89 (2016)
36. Jackowski, Z.: A new characterization of proper interval graphs. Discrete Math. **105**(1–3), 103–109 (1992)
37. Jacobson, M.S., Peters, K.: Complexity questions for n-domination and related parameters. Congr. Numer. **68**, 7–22 (1989). Eighteenth Manitoba Conference on Numerical Mathematics and Computing (Winnipeg, MB, 1988)
38. Kang, D.Y., Kwon, O.-J., Strømme, T.J.F., Telle, J.A.: A width parameter useful for chordal and co-comparability graphs. Theoret. Comput. Sci. **704**, 1–17 (2017)
39. Kazemi, A.P.: On the total k-domination number of graphs. Discuss. Math. Graph Theory **32**(3), 419–426 (2012)
40. Keil, J.M.: Total domination in interval graphs. Inform. Process. Lett. **22**(4), 171–174 (1986)
41. Kulli, V.R.: On n-total domination number in graphs. Graph Theory. Combinatorics, Algorithms, and Applications (San Francisco, CA, 1989), pp. 319–324. SIAM, Philadelphia (1991)
42. Lan, J.K., Chang, G.J.: Algorithmic aspects of the k-domination problem in graphs. Discrete Appl. Math. **161**(10–11), 1513–1520 (2013)
43. Lan, J.K., Chang, G.J.: On the algorithmic complexity of k-tuple total domination. Discrete Appl. Math. **174**, 81–91 (2014)
44. Lee, C.-M., Chang, M.-S.: Variations of Y-dominating functions on graphs. Discrete Math. **308**(18), 4185–4204 (2008)
45. Liao, C.-S., Chang, G.J.: k-tuple domination in graphs. Inform. Process. Lett. **87**(1), 45–50 (2003)
46. Liao, C.-S., Lee, D.T.: Power domination in circular-arc graphs. Algorithmica **65**(2), 443–466 (2013)
47. Liedloff, M., Kloks, T., Liu, J., Peng, S.-L.: Efficient algorithms for Roman domination on some classes of graphs. Discrete Appl. Math. **156**(18), 3400–3415 (2008)
48. Lin, C.-J., Liu, J.-J., Wang, Y.-L.: Finding outer-connected dominating sets in interval graphs. Inform. Process. Lett. **115**(12), 917–922 (2015)
49. Looges, P.J., Olariu, S.: Optimal greedy algorithms for indifference graphs. Comput. Math. Appl. **25**(7), 15–25 (1993)
50. Oum, S.-I., Seymour, P.: Approximating clique-width and branch-width. J. Combin. Theory Ser. B **96**(4), 514–528 (2006)
51. Panda, B.S., Paul, S.: A linear time algorithm for liar's domination problem in proper interval graphs. Inform. Process. Lett. **113**(19–21), 815–822 (2013)

52. Panda, B.S., Pradhan, D.: A linear time algorithm to compute a minimum restrained dominating set in proper interval graphs. Discrete Math. Algorithms Appl. **7**(2), 1550020 (2015). 21
53. Pradhan, D.: Algorithmic aspects of k-tuple total domination in graphs. Inform. Process. Lett. **112**(21), 816–822 (2012)
54. Pramanik, T., Mondal, S., Pal, M.: Minimum 2-tuple dominating set of an interval graph. Int. J. Comb. 14 (2011). Article ID 389369
55. Ramalingam, G., Pandu, C.: Total domination in interval graphs revisited. Inform. Process. Lett. **27**(1), 17–21 (1988)
56. Ramalingam, G., Rangan, C.P.: A unified approach to domination problems on interval graphs. Inform. Process. Lett. **27**(5), 271–274 (1988)
57. Roberts, F.S.: Indifference graphs. In: Proof Techniques in Graph Theory (Proceedings of Second Ann Arbor Graph Theory Conference, Ann Arbor, Michigan, 1968), pp. 139–146. Academic Press, New York (1969)
58. Chiarelli, N., Hartinger, T.R., Leoni, V.A., Lopez Pujato, M.I., Milanič, M.: New algorithms for weighted k-domination and total k-domination problems in proper interval graphs. arXiv:1803.04327 [cs.DS] (2018)