Asian-European Journal of Mathematics
(2021) 2150049 (12 pages)

DOI: 10.1142/S1793557121500492

An optimal algorithm to find minimum k-hop connected dominating set of permutation graphs

Amita Samanta Adhya*

Research Centre in Natural Sciences
(Department of Computer Science)
Raja N. L. Khan Women's College (Autonomous)
Midnapore 721 102, India
amitasamanta1@gmail.com

Sukumar Mondal

Department of Mathematics
Raja N. L. Khan Women's College (Autonomous)
Midnapur 721 102, India
sm5971@rediffmail.com

Sambhu Charan Barman

Department of Mathematics
Shahid Matangini Hazra Government College for Women
Purba Medinipur 721649, India
barman.sambhu@gmail.com

Communicated by B. K. Dass
Received November 17, 2019
Revised December 26, 2019
Accepted January 23, 2020
Published March 25, 2020

A set $D_k \subseteq V$ is said to be a k-hop dominating set (k-HDS) of a graph $G = (V, E)$ if every vertex $x \in V$ is within k-distances from at least one vertex $t \in D_k$, i.e. $d(x, t) \leq k$, where k is a fixed positive integer. A dominating set D_k is said to be minimum k-hop connected dominating set of a graph G, if it is minimal as well as it is k-HDS and the subgraph of G made by D_k is connected. In this paper, we present an $O(n)$-time algorithm for computing a minimum k-hop connected dominating set of permutation graphs with n vertices.

Keywords: Algorithms; k-hop connected domination; permutation graphs.

AMS Subject Classification: 05C30, 05C12, 68R10, 68Q25

*Corresponding author.

1. Introduction

A graph $G = (V, E)$ with vertex set $V = \{1, 2, 3, \ldots, n\}$ is called a *permutation graph* iff there exists a permutation $\pi = \{\pi(1), \pi(2), \ldots, \pi(n)\}$ on V such that for all i_1, $i_2 \in V$, $(i_1, i_2) \in E$ if and only if $(i_1 - i_2)\{\pi^{-1}(i_1) - \pi^{-1}(i_2)\} < 0$, where for each $i_1 \in V$, $\pi^{-1}(i_1)$ denotes the position of the number i_1 in π [24]. The concept of permutation graphs is introduced by Adin and Roichman. Permutation graph is a sub-class of intersection graphs [24] and comparability graphs [51]. A permutation graph can also be created by its corresponding matching diagram which contains two horizontal parallel lines, one is top line and the other is bottom line. A permutation graph and its matching diagram are shown in Fig. 1.

The matching diagram of a permutation graph can be formed in $O(n^2)$ time only when the graph is given in the form of an adjacency list/adjacency matrix [24, 46]. For this reason, we presume that a permutation representation is given for the input graph. Also, we create two arrays: one is $\pi(j)$, $j = 1, 2, 3, \ldots, n$ used to store the permutation numbers and other array is $\pi^{-1}(j)$, $j = 1, 2, 3, \ldots, n$ used to store the inverse permutation of π. We can compute the array $\pi^{-1}(j)$ from array π in $O(n)$ time.

1.1. *Some definitions*

For a graph $G = (V, E)$, a vertex dominates itself and all of its adjacent vertices. A set $D \subseteq V$ is called a *dominating set* of a graph G if every vertex in V is dominated by at least one vertex in D. A dominating set D is called minimal if there does not exist any $D' \subset D$ such that D' is also a dominating set of the same graph. Various types of domination like connected domination [21], k-hop domination [23, 38, 56], k-tuple domination [40, 41, 52, 55], weighted domination [11], edge domination [43, 49, 59], independent domination [12], locating domination [44], total/open domination [10, 20, 25, 30, 31] and paired-domination [16–19, 32, 35, 37, 39, 50, 53] have been studied in the literature, see [13, 27, 28, 33]. A set

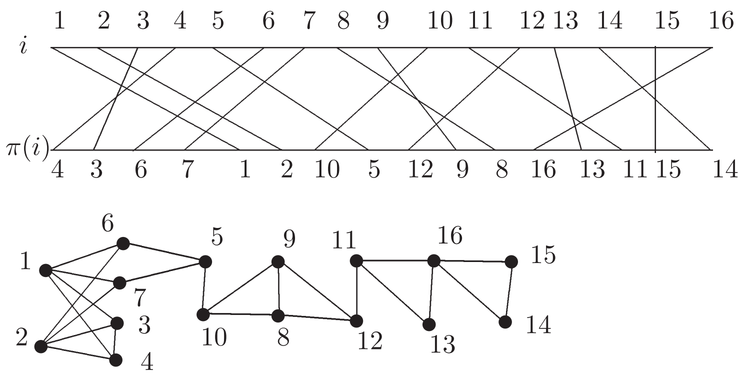

Fig. 1. A permutation graph G and its matching diagram.

$D_k \subseteq V$ is said to be a k-hop dominating set (k-HDS) of a graph $G = (V, E)$ if every vertex $x \in V$ is within k-distances from at least one vertex $t \in D_k$, i.e. $d(x, t) \leq k$, where k is a fixed positive integer. For the particular case $k = 1$, 1-hop dominating set is named as dominating set. Slater mentions a k-HDS as a k-basis [56] and develops the k-basis problem in a more general form called R-dominating set. Now, a dominating set D_k is known to be minimum k-hop connected dominating set of a graph G, if it is minimal as well as it is k-HDS and the subgraph of G made by D_k is connected.

1.2. *Applications*

Domination is a crucial problem in graph theory. There are several applications (see [1, 6–9]) of domination in the real world such as radio stations, modeling of biological networks, facility location problems, land surveying, computer communication networks, problems of locating radar stations, school bus routing, coding theory and kernels of games, etc. Though there are lot of applications of k-hop domination in the real life, an explanation in terms of communication networks will be used here. If V represents a set of cities and an edge represents a communication link, then, as in [42], everybody wants to select a minimum number of cities as sites for transmitting stations so that every city either contains a transmitter or can receive messages from at least one of the transmitting stations through the links. If only direct transmissions are acceptable, then one wishes to compute a minimum 1-hop dominating set. If communication over paths of k links (but not of $k + 1$ links) is adequate in quality and rapidity, the problem becomes that of computing a k-HDS with minimum number vertices. For enhancing the performance of efficiency in communication, a connected dominating set is used as the virtual backbone of wireless networks.

1.3. *Survey of the related works*

In graph theory, domination problems have been researched deeply by many researchers. In 1862, the chess master de Jaenisch wrote a paper [22] on the applications of mathematical analysis on chess. Chang [13] has broadly studied about domination in graphs. Furthermore, various types of domination can be found in [14, 20, 25–31, 57]. Among the various forms of domination, there are some brief discussion on the k-hop domination in the literature, see [2, 13, 25, 27, 28, 32, 33, 43, 47, 53, 59]. Demaine *et al.* [23] studied the class of planar graph (and some other graphs also) G and composed an algorithm for computing an optimal k-HDS in $O(n^4)$ time, where $|V(G)| = n$. Basuchowdhuri *et al.* [5] set up influential nodes for social network using k-HDS. They also proved that decision version of the k-HDS problem is NP-complete. Henning *et al.* [34] proved that the problem on hop dominating set is NP-complete for planar chordal graphs and planar bipartite graphs. Also, Farhadi *et al.* [36] ascertained that the problem on k-HDS is NP-complete for planar chordal graphs and planar bipartite graphs. Natarajan *et al.* [47] studied the equality of hop domination number and other parameters on domination such as connected domination number and total domination number on complete graphs, Petersen graphs, complete bipartite graphs, cycles, paths and wheels. Furthermore, Ayyaswamy *et al.* [2] observed the bounds on the hop domination number of a tree. Rana *et al.* [54] have presented an efficient algorithm for solving the distance k-domination problem on permutation graphs. Kundu *et al.* [38] gave an $O(n)$ time algorithm for discovering an optimal k-HDS of a tree. Recently, Barman *et al.* [4] presented an optimal algorithm to compute a minimum k-HDS of interval graphs in $O(n)$ time.

1.4. *Main outcome*

In our paper, we present an $O(n)$-time algorithm to compute a minimum k-hop connected dominating set of undirected and connected permutation graph with n vertices. We compute analytically a connected dominating set but it varies in terms of number of hops essential to dominate all the nodes. We have generalized the matter as k-hop connected dominating set problem on permutation graphs.

1.5. *Organization of the paper*

In Sec. 2, we recapitulate about the formation of BFS-tree $T^*(1)$ of permutation graphs. Some notations concerned to our problem are also presented in this section. In Sec. 3, we prove some vital results concerned to the k-hop connected dominating set of permutation graphs. An algorithm is also presented for finding a minimum k-hop connected dominating set of permutation graphs in Sec. 4. Also, the time and space complexity are calculated in this section. The conclusion is written in Sec. 5.

2. BFS-Tree and Principal Path on It

In graph theory, breadth first search (BFS) is a well-known and crucial graph traversal technique. Also, it can set up a BFS-tree. In BFS, it started with vertex x as root; at first, we visit those vertices which are adjacent to x. After then, for each of the adjacent vertices of x, we visit their unvisited adjacent vertices. We carry on until all vertices within reach from the root x have been visited.

To the best of our knowledge, the time complexity to form a BFS-tree on general graphs is $O(n+m)$ [58]. A BFS algorithm on tree is presented by Chen *et al.* [15]. Besides these, Olariu [48] and Mondal *et al.* [45] formed BFS tree, respectively, on interval and trapezoid graphs in $O(n)$-time. Also, Barman *et al.* [3] designed an *Algorithm PBFS* to form a BFS tree $T^*(x)$ taking root as x on the permutation graph in $O(n)$-time. The BFS tree $T^*(1)$ rooted at 1 of the permutation graph of Fig. 1 is displayed in Fig. 2. We take the level of a vertex u as a distance of u from the root 1 of the tree $T^*(1)$ and denoted by $level(u), u \in V$ and take the level of root 1 as 0. The level of each vertex on BFS-tree $T^*(1)$, $1 \in V$ can be determined by the BFS algorithm of Chen and Das [15].

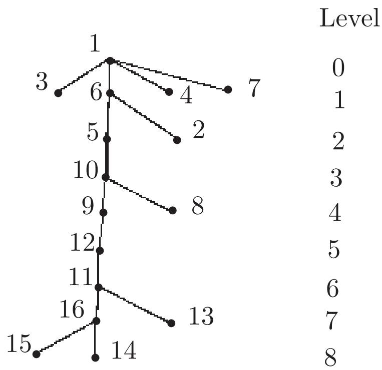

Fig. 2. BFS-tree $T^*(1)$ of permutation graph G of Fig. 1.

Now, for the BFS-tree $T^*(1)$ taking root as 1, let w be any vertex at level h, where h is the height of the BFS-tree $T^*(1)$. We presume that the shortest path between 1 and w which is $w \to \text{parent}(w) \to \text{parent}(\text{parent}(w)) \to \cdots \to 1$ as the *principal path* between 1 and w. Furthermore, the vertex on the principal path at level i is denoted by v_i^*, $i = 0, 1, \ldots, h$.

2.1. *Some notations*

Here, we present some notations which are essential throughout the paper.

v_i^* : v_i^* is the vertex on the principal path of the BFS-tree $T^*(1)$ at level i.
$N(v)$: open neighborhood set of the vertex v, i.e. $N(v) = \{x : (x, v) \in E\}$.
$N[v]$: closed neighborhood set of the vertex v, i.e. $N[v] = N(v) \cup \{v\}$.
M_i : M_i is the set of vertices at level i on the BFS-tree $T^*(1)$.
X_i : $X_i = M_i - \{v_i^*\}$.
h : the height of the BFS-tree $T^*(1)$.
$d(x, y)$: the shortest distance between two vertices x and y.
D_k : k-hop connected dominating set.

3. Important Results Relating to D_k of Permutation Graphs

Lemma 1. *If $X_1 = \phi$ or if $(x, v_1^*) \in E$, for all $x \in M_1$, then v_k^* is a possible member of D_k.*

Proof. Let x_i be any member of the set X_i, for $i = 0, 1, 2, \ldots, h$. So, $d(v_k^*, v_0^*) = k$ (as $v_0^* \to v_1^* \to v_2^* \to \cdots \to v_k^*$). Now, if $X_1 = \phi$, then $d(v_k^*, x_i) \le k + 2 - i$ (as $x_i \to v_{i-1}^* \to v_i^* \to v_{i+1}^* \to \cdots \to v_k^*$ or $x_i \to v_i^* \to v_{i+1}^* \to v_{i+2}^* \to \cdots \to v_k^*$ or $x_i \to v_{i+1}^* \to v_{i+2}^* \to \cdots \to v_k^*$), for $i = 2, 3, \ldots, (k-1)$. Furthermore, $d(v_k^*, x_k) \le 2$ (as $x_k \to v_{k-1}^* \to v_k^*$ or $x_k \to v_k^*$). Therefore, $d(v_k^*, x) \le k$, for all $x \in \bigcup_{i=0}^{k} M_i$.

Again, if $(x, v_1^*) \in E$, for all $x \in M_1$, then $d(v_k^*, x_1) \leq k$ (as $x_1 \to v_1^* \to v_2^* \to v_3^* \to \cdots \to v_k^*$ or $x_1 \to v_2^* \to v_3^* \to \cdots \to v_k^*$). Also, $d(v_k^*, x_i) \leq k + 2 - i$ (as $x_i \to v_{i-1}^* \to v_i^* \to v_{i+1}^* \to \cdots \to v_k^*$ or $x_i \to v_i^* \to v_{i+1}^* \to v_{i+2}^* \to \cdots \to v_k^*$ or $x_i \to v_{i+1}^* \to v_{i+2}^* \to \cdots \to v_k^*$), for $i = 2, 3, \ldots, (k-1)$ and $d(v_k^*, x_k) \leq 2$ (as $x_k \to v_{k-1}^* \to v_k^*$ or $x_k \to v_k^*$). Therefore, $d(v_k^*, x) \leq k$, for all $x \in \bigcup_{i=0}^{k} M_i$. Therefore, v_k^* is a possible member of D_k. □

Lemma 2. *If at least one member of $M_1 - \{v_1^*\}$ is not adjacent with v_1^* and if $(x, v_2^*) \in E$, for all $x \in M_1$ then v_k^* is a possible member of D_k.*

Proof. Let x_i be any member of the set X_i, for $i = 0, 1, 2, \ldots, h$ and y be a member of X_1 such that $(y, v_1^*) \notin E$ and $(y, v_2^*) \in E$. Since parent$(y) = v_0^*$, $d(v_k^*, y) = k - 1$ (as $y \to v_2^* \to v_3^* \to \cdots \to v_k^*$). Also, $d(v_k^*, v_0^*) = k$ (as $v_0^* \to v_1^* \to v_2^* \to \cdots \to v_k^*$) and $d(v_k^*, x_1) \leq k$ (as $x_1 \to v_1^* \to v_2^* \to \cdots \to v_k^*$ or $x_1 \to v_2^* \to v_3^* \to \cdots \to v_k^*$), for all $x_1 \in X_1$. Now, $d(v_k^*, x_i) \leq k + 2 - i$ (as $x_i \to v_{i-1}^* \to v_i^* \to v_{i+1}^* \to \cdots \to v_k^*$ or $x_i \to v_i^* \to v_{i+1}^* \to \cdots \to v_k^*$ or $x_i \to v_{i+1}^* \to v_{i+2}^* \to \cdots \to v_k^*$), for $i = 2, 3, \ldots, (k-1)$. Furthermore, $d(v_k^*, x_k) \leq 2$ (as $x_k \to v_{k-1}^* \to v_k^*$ or $x_k \to v_k^*$). Hence, $d(v_k^*, x) \leq k$, for all $x \in \bigcup_{i=0}^{k} M_i$. Therefore, v_k^* is a possible member of D_k. □

Lemma 3. *For all $x \in \bigcup_{i=0}^{k-1} M_i$, $d(v_{k-1}^*, x) \leq k$.*

Proof. Let x_i be any member of the set X_i, for $i = 0, 1, 2, \ldots, h$. So, $d(v_{k-1}^*, v_0^*) = k - 1 \leq k$ (as $v_0^* \to v_1^* \to v_2^* \to \cdots \to v_{k-1}^*$) and $d(v_{k-1}^*, x_1) \leq k$ (as $x_1 \to v_0^* \to v_1^* \to v_2^* \to \cdots \to v_{k-1}^*$ or $x_1 \to v_1^* \to v_2^* \to \cdots \to v_{k-1}^*$ or $x_1 \to v_2^* \to v_3^* \to \cdots \to v_{k-1}^*$). Now, $d(v_{k-1}^*, x_i) \leq k + 1 - i$ (as $x_i \to v_{i-1}^* \to v_i^* \to v_{i+1}^* \to \cdots \to v_{k-1}^*$ or $x_i \to v_i^* \to v_{i+1}^* \to \cdots \to v_{k-1}^*$ or $x_i \to v_{i+1}^* \to v_{i+2}^* \to \cdots \to v_{k-1}^*$), for $i = 2, 3, \ldots, (k-2)$. Furthermore, $d(v_{k-1}^*, x_{k-1}) \leq 2$ (as $x_{k-1} \to v_{k-2}^* \to v_{k-1}^*$ or $x_{k-1} \to v_{k-1}^*$). Therefore, $d(v_{k-1}^*, x) \leq k$, for all $x \in \bigcup_{i=0}^{k-1} M_i$. □

Corollary 1. *v_{k-1}^* is always a possible member of D_k.*

Lemma 4. *If at any stage, v_l^* is a member of D_k, then $d(v_l^*, x) \leq k$, for all $x \in \bigcup_{i=1}^{k} M_{l+i}$.*

Proof. Let at any stage, v_l^* be a member of the set D_k and x_i be any member of the set M_i, for $i = 0, 1, 2, \ldots, h$. We know that, parent$(x_i) = v_{i-1}^*$, for $i = 1, 2, 3, \ldots, h$. So, $d(v_l^*, x_{l+i}) \leq i \leq k$ (as $v_l^* \to v_{l+1}^* \to v_{l+2}^* \to \cdots \to v_{l+i-1}^* \to x_{l+i}$), for $i = 1, 2, \ldots, k$. So, $d(v_l^*, x) \leq k$, for all $x \in \bigcup_{i=1}^{k} M_{l+i}$. □

Lemma 5. *For all $x \in \bigcup_{i=h-k+1}^{h} M_i$, $d(v_{h-k}^*, x) \leq k$, where $h \geq k$.*

Proof. Let x_i be any member of the set M_i, for $i = 0, 1, 2, \ldots, h(\geq k)$. We know that, parent$(x_i) = v_{i-1}^*$, for $i = 1, 2, 3, \ldots, h$. So, $d(v_{h-k}^*, x_{h-k+i}) = i \leq k$

(as $v^*_{h-k} \to v^*_{h-k+1} \to \cdots \to v^*_{h-k+i-1} \to x_{h-k+i}$), for $i = 1, 2, 3, \ldots, k$. Therefore, $d(v^*_{h-k}, x) \leq k$, for all $x \in \bigcup_{i=1}^{k} M_{h-k+i}$. □

Corollary 2. *v^*_{h-k} is always a possible member of D_k, where $h \geq k$.*

Lemma 6. *For all $x \in \bigcup_{i=0}^{2k-1} M_i$, $d(v^*_{k-1}, x) \leq k$.*

Proof. Let x_i be any member of the set M_i, for $i = 0, 1, 2, \ldots, h$. Now, by Lemma 3, $d(v^*_{k-1}, x) \leq k$, for all $x \in \bigcup_{i=0}^{k-1} M_i$. Again, we know that, parent$(x_i) = v^*_{i-1}$, for $i = 1, 2, 3, \ldots, h$. So, $d(v^*_{k-1}, x_{k+i}) = i+1$ (as $v^*_{k-1} \to v^*_k \to v^*_{k+1} \to \cdots \to v^*_{k+i-1} \to x_{k+i}$), for $i = 0, 1, 2, 3, \ldots, k-1$. Hence, $d(v^*_{k-1}, x) \leq k$, for all $x \in \bigcup_{i=0}^{2k-1} M_i$. □

Lemma 7. *If $h < k$, then $D_k = \{v^*_i\}$, where i is any one of the set $\{0, 1, 2, \ldots, h\}$.*

Proof. Let $h < k$ and x_i be any member of the set X_i, for $i = 0, 1, 2, \ldots, h$. Now, by Lemma 4, $d(v^*_0, x) \leq h < k$, for all $x \in \bigcup_{i=1}^{h} M_i$. Hence, v^*_0 is a possible member of D_k . Now, we show that v^*_h is also a possible member of D_k. We know, $d(v^*_h, v^*_0) = h < k$ (as $v^*_0 \to v^*_1 \to \cdots \to v^*_h$). Also, $d(v^*_h, x_i) \leq (h + 2 - i) \leq k$ (as $x_i \to v^*_{i-1} \to v^*_i \to \cdots \to v^*_h$ or $x_i \to v^*_i \to v^*_{i+1} \to \cdots \to v^*_h$ or $x_i \to v^*_{i+1} \to v^*_{i+2} \to \cdots \to v^*_h$), for $i = 1, 2, 3, \ldots, h$. So, $d(v^*_h, x) \leq k$, for all $x \in \bigcup_{i=0}^{h} M_i$. Hence, v^*_h is also a possible member of D_k. Therefore, if $h < k$, then v^*_i be a possible member of D_k, where i is any member of the set $\{0, 1, 2, \ldots, h\}$. □

Lemma 8. *If $k \leq h \leq (2k-1)$, then $D_k = \{v^*_{k-1}\}$.*

Proof. Let x_i be any member of the set M_i, for $i = 0, 1, 2, \ldots, h$. We know, by Lemma 6, $d(v^*_{k-1}, x) \leq k$, for all $x \in \bigcup_{i=0}^{2k-1} M_i$. Therefore, if $k \leq h \leq (2k-1)$, then $D_k = \{v^*_{k-1}\}$. □

4. Algorithm and its Complexity

All probable cases for selecting the members of D_k are formerly presented in terms of lemmas. We have noticed that the last possible member of D_k will be v^*_{h-k}, $h \geq k$. Also, the vertex v^*_{k-1} can be considered as a member of D_k at any situation (by Corollary 1). Furthermore, v^*_k may be a possible member of D_k, depends upon some conditions discussed in Lemmas 1 and 2. But our target is to compute the set D_k in such a way that $|D_k|$ must be minimum. Now, we present our proposed algorithm for finding a minimum k-hop connected dominating set D_k of permutation graphs:

Algorithm MKHCDS
Input: A permutation representation $i, \pi(i)$ of a permutation graph G, for $i = 1, 2, 3, \ldots, n$.
Output: Minimum k-hop connected dominating set D_k of the permutation graph G.

Step 1: Construct the BFS-tree $T^*(1)$ with root at 1 and initially, let $D_k = \phi$ and $v_0^* = 1$.
Step 2: Compute the vertices on the principal path of the BFS-tree $T^*(1)$ and let them be $v_i^*, i = 0, 1, 2, 3, \ldots, h$.
Step 3: Compute the sets $M_i, i = 0, 1, 2, 3, \ldots, h$.
Step 4: If $h < k$
then $D_k = D_k \cup \{v_l^*\}$,
where l is any one member of the set $\{0, 1, 2, 3, \ldots, h\}$.
//Lemma 7//
else if $k \le h \le 2k - 1$
then $D_k = D_k \cup \{v_{k-1}^*\}$. //Lemma 8//
else if $X_1 = \phi$ or $(x, v_1^*) \in E$ or $(x, v_2^*) \in E$, for all $x \in M_1$
then $D_k = D_k \cup \{v_k^*, v_{k+1}^*, \ldots, v_{h-k}^*\}$. //Lemma 2 and Corollary 2//
else $D_k = D_k \cup \{v_{k-1}^*, v_k^*, \ldots, v_{h-k}^*\}$. //Corollaries 1 and 2//
end if
end MKHCDS.

Using **Algorithm MKHCDS**, we get a minimum 4-hop connected dominating set $D_4 = \{10, 9\}$ of the permutation graph G of Fig. 1.

Lemma 9. *The set D_k is a minimum k-hop connected dominating set of permutation graph.*

Proof. In Algorithm MKHCDS, if $h < k$ then, by Lemma 7, we set $D_k = \{v_l^*\}$, where l is any one member of the set $\{0, 1, 2, 3, \ldots, h\}$. So, in that case, D_k is minimum and connected. Again, if $k \le h \le 2k - 1$, then, by using Lemma 8, we set $D_k = \{v_{k-1}^*\}$. Therefore D_k is minimum and connected for $k \le h \le 2k - 1$. Moreover, when $h \ge 2k$, there are two possible vertices v_k^* (using Lemma 2) or v_{k-1}^* (using Corollary 1) for selecting the first member of D_k. Here v_k^* dominates $|\bigcup_{i=0}^{2k} M_i|$ vertices and v_{k-1}^* dominates $|\bigcup_{i=0}^{2k-1} M_i|$ vertices. Also, $|\bigcup_{i=0}^{2k} M_i| > |\bigcup_{i=0}^{2k-1} M_i|$. For this reason, we select v_k^* as the first member of D_k, if possible. If it is not possible, only then we select v_{k-1}^* as the first member of D_k. Again, we select v_{h-k}^*, by Corollary 2, as a last member of D_k as $d(v_{h-k}^*, x) = k$, for all $x \in M_h$. The other member of D_k (excluding v_k^* (or v_{k-1}^*) and v_{h-k}^*) are selected in such a way that each vertex of V is within k steps from at least one member of D_k. So, D_k is minimum k-HDS. Now, we have to prove that D_k is connected. Since all the members of D_k lie on the principal path of the BFS-tree $T^*(1)$ of the permutation graph G and we know that tree is a connected graph. So, the graph (which is a path) induced by the members of D_k is a subgraph of the permutation graph G. Hence, D_k is connected. Therefore, D_k is a minimum k-hop connected dominating set of permutation graph. □

Theorem 1. *The time complexity and space complexity for computing a minimum k-hop connected dominating set D_k of a permutation graph are same and equals to $O(n)$, where $|V| = n$.*

Proof. In Step 1, the time taken to construct the BFS-tree $T^*(1)$ is $O(n)$ [3]. Since the principal path contains only $h+1$ vertices, so the members of principal path (identified at Step 2) can be computed in $O(n)$ time. In Step 3, i.e. the sets M_i, $i = 0, 1, 2, \ldots, h$ can be calculated in $O(n)$ time as M_i, $i = 0, 1, 2, \ldots, h$ are mutually exclusive. In Step 4, if $h < k$ or $k \leq h \leq (2k-1)$, then D_k can be computed in constant time as $|D_k| = 1$. On the other hand, if $h \geq 2k$, then time complexity for finding v_k^*(or v_{k-1}^*) is $O(2|M_1|) < O(n)$. Hence, Step 4 can be finished in $O(n)$ time as $|D_k| \leq n$. Therefore, overall time complexity of algorithm MKHCDS is $O(n)$ time.

Now, in permutation representation of a permutation graph, there are two sets: one is $V = \{1, 2, \ldots, n\}$ and other set is $\pi(i)$, $i = 1, 2, \ldots, n$. So, one can store these two sets in $O(n)$ space. Also, for BFS-tree $T^*(1)$, we can store the sets M_i, $i = 0, 1, 2, \ldots, h$ in $O(n)$ space as M_i, $i = 0, 1, 2, \ldots, h$ are pairwise mutually exclusive sets. Furthermore, minimum k-hop connected dominating set D_k can be stored in $O(n)$ space as $|D_k| \leq n$. Hence the total space complexity of Algorithm MKHCDS is $O(n)$. □

5. Conclusion

In graph theory, k-hop domination has lot of applications in real world. Here, we design an optimal algorithm for computing a minimum k-hop connected dominating set of simple, undirected and connected permutation graphs $G(V, E)$ in $O(n)$ time, where $|V| = n$. We realize that it would be engrossing to design an optimal algorithm to find a minimum k-hop connected dominating set of trapezoid graphs, circular-arc graphs and unit disc graphs, etc.

References

1. B. Awerbuch and D. Peleg, Sparse partitions (extended abstract), *in 31st IEEE Annual Symp. on Foundations of Computer Science* (1990), pp. 503–513.
2. S. K. Ayyaswamy, B. Krishnakumari, C. Natarajan and Y. B. Venkatakrishnan, Bounds on the hop domination number of a tree, *Proc. Math. Sci. Indian Acad. Sci.* **125** (2015) 449–455.
3. S. C. Barman, M. Pal and S. Mondal, An efficient algorithm to find next-to-shortest path on permutation graphs, *J. Appl. Math. Comput.* **31** (2009) 369–384.
4. S. C. Barman, M. Pal and S. Mondal, An optimal algorithm to find minimum k-hop dominating set of interval graphs, *Discrete Math. Algorithms Appl.* **11**(2) (2019) 1950016.
5. P. Basuchowdhuri and S. Majumder, Finding influential nodes in social networks using minimum k-hop dominating set, in *Proc. Int. Conf. Applied Algorithms*, Vol. 8321 (Springer, 2014), pp. 137–151.

6. D. A. Benson, M. S. Boguski, D. J. Lipman, J. Ostell, B. F. Ouellette, B. A. Rapp and D. L. Wheeler, Nucleic acids research, *Gen Bank* **27**(1) (1999) 12–17.
7. S. Berchtold, C. Bohm and H. P. Kriegel, The pyramid-technique: Towards breaking the curse of dimensionality, in *Proc. International Conference on Management of Data* (ACM SIGMOD, Seattle, Washington, 1998), pp. 142–153.
8. S. Berchtold, D. Keim and H. P. Kriegel, The X-tree: An index structure for high dimensional data, in *Proceedings of 22nd Conf. on Very Large Databases*, Mumbai (Bombay), India, (Morgan Kaufmann Publishers Inc., United States, 1996), pp. 28–39.
9. E. Berger, Dynamic monopolies of constant size, *Technical Report, Los Alamos National Laboratories* (1999).
10. A. A. Bertossi and A. Gori, Total domination and irredundance in weighted interval graphs, *SIAM J. Discrete Math.* **1**(3) (1988) 317–327.
11. M. S. Chang, Weighted domination of cocomparability graphs, *Discrete Appl. Math.* **80**(2–3) (1997) 135–148.
12. G. J. Chang, The weighted independent domination problem is NP-complete for chordal graphs, *Discrete Appl. Math.* **143**(1–3) (2004) 351–352.
13. G. J. Chang, Algorithmic aspect of domination in graphs, in *Handbook of Combinatorial Optimization*, 2nd edn. (Springer-Verlag, Newyork, 2013), pp. 339–405.
14. H. S. Chao, F. R. Hsu and R. C. T. Lee, An optimal algorithm for finding the minimum cardinality dominating set on permutation graphs, *Discrete Appl. Math.* **102** (2000) 159–173.
15. C. C. Y. Chen and S. K. Das, Breadth-first traversal of trees and integer sorting in parallel, *Inf. Process. Lett.* **41** (1992) 39–49.
16. L. Chen, C. Lu and Z. Zeng, A linear time algorithm for paired-domination problems in strongly chordal graphs, *Inf. Process. Lett.* **110**(1) (2009) 20–23.
17. L. Chen, C. Lu and Z. Zeng, Labelling algorithm for paired-domination problems in block and interval graphs, *J. Comb. Optim.* **19**(4) (2010) 457–470.
18. T. C. E. Cheng, L. Y. Kang and C. T. Ng, Paired domination on interval and circular-arc graphs, *Discrete Appl. Math.* **155**(16) (2007) 2077–2086.
19. T. C. E. Cheng, L. Y. Kang and E. Shan, A polynomial-time algorithm for the paired domination problem on permutation graphs, *Discrete Appl. Math.* **157**(2) (2009) 262–271.
20. E. Cockayne, R. Dawes and S. Hedetniemi, Total domination in graphs, *Networks* **10**(3) (1980) 211–219.
21. A. D'Atri and M. Moscarini, Distance-hereditary graphs, Steiner trees, and connected domination, *SIAM J. Comput.* **17**(3) (1988) 521–538.
22. C. F. De Jaenisch, *Trait des Applications de l'Analyse Mathematique au Jeu des Echecs* (Petrograd, 1862).
23. E. D. Demaine, M. Hajjaghay and D. M. Thilikos, Fixed parameter algorithm for (k, r) center in planar graphs and map graphs, *ACM Trans. Algorithm* **1**(1) (1862) 1–16.
24. M. C. Golumbic, *Algorithmic Graph Theory and Perfect Graphs* (Academic Press, Newyork, 2004).
25. T. W. Haynes, S. T. Hedetniemi and P. J. Slater, *Domination in Graphs*: *Advanced Topics* (Marcel Dekker. Inc., Newyork, 1998).
26. T. W. Haynes, S. T. Hedetniemi and P. J. Slater, *Domination in Graphs*: *Selected Topics* (Marcel Dekker, Newyork, 1998).
27. T. W. Haynes, S. T. Hedetniemi and P. J. Slater, *Domination in Graphs: The Theory* (Marcel Dekker, Newyork, 1998).

28. T. W. Haynes, S. T. Hedetniemi and P. J. Slater, *Fundamentals of Domination in Graphs* (Marcel Dekker, Newyork, 1998).
29. T. W. Haynes, C. M. Mynhardt and L. C. van der Merwe, Riticality index of total domination, *Congr. Numer.* **131** (1998) 67–73.
30. T. W. Haynes, C. M. Mynhardt and L. C. van der Merwe, Total domination edge critical graphs, *Utilitas Math.* **54** (1998) 229–240.
31. T. W. Haynes, C. M. Mynhardt and L. C. van der Merwe, Total domination edge critical graphs with maximum diameter, *Discuss. Math. Graph Theory* **21** (2001) 187–205.
32. T. W. Haynes and P. J. Slater, Paired-domination in graphs, *Networks* **32** (1998) 199–206.
33. S. T. Hedetniemi and R. C. Laskar (eds.), Topics on Domination, *Annals of Discrete Mathematics*, Vol. 48 (North-Holland, Amsterdam, 1991).
34. M. A. Henning and N. Jafari Rad, On 2-Step and Hop Dominating Sets in Graphs, *Graphs Combin.* **33** (2017) 913–927.
35. R. W. Hung, Linear time algorithm for the paired-domination problem in convex bipartite graphs, *Theory Comput. System* **50**(4) (2012) 721–738.
36. M. F. Jalavand and N. Jafari, On the complexity of k-step and k-hop dominating sets in graphs, *Math. Montisnigri Math.* **12** (2017) 36–41.
37. L. Kang, M. Y. Sohn and T. C. E. Cheng, Paired-domination in inflated graphs, *Theory Compute. Syst.* **320**(2–3) (2004) 485–494.
38. S. Kundu and S. Majumder, A linear time algorithm for optimal k-hop dominating set of a tree, *Inf. Process. Lett.* **116** (2016) 197–202.
39. E. Lappas, S. D. Nicolopoulos and L. Palios, *An O(n)-Time Algorithm for the Paired-Domination Problem on Permutation Graphs*, Lecture Notes in Computer Science, Vol. 5874 (Springer, 2009), pp. 368–379, ISBN 978-3-642-10216-5.
40. C. S. Liao and G. J. Chang, Algorithmic aspect of k-tuple domination in graphs, *Taiwanese J. Math.* **6** (2002) 415–420.
41. C. S. Liao and G. J. Chang, k-tuple domination in graphs, *Inf. Process. Lett.* **87** (2003) 45–50.
42. C. Liu, *Introduction to Combinatorial Mathematics* (McGraw-Hill, Newyork, 1968).
43. C. L. Lu, M. T. Ko and C. Y. Tang, Perfect edge domination and efficient edge domination in graphs, *Discrete Appl. Math.* **119** (2002) 227–250.
44. J. McCoy and M. A. Henning, Locating and paired-dominating sets in graphs, *Discrete Appl. Math.* **157**(15) (2009) 3268–3280.
45. S. Mondal, M. Pal and T. K. Pal, An optimal algorithm for solving all-pairs shortest paths on trapezoid graphs, *Intern. J. Comput. Eng. Sci.* **3**(2) (2002) 103–116.
46. J. H. Muller and J. Spinrad, Incremental modular decomposition, *J. ACM* **36** (1989) 1–19.
47. C. Natarajan and S. K. Ayyaswamy, Hop Domination in Graphs-II, *An. Stt. Univ. Ovidius Constanta* **23**(2) (2015) 187–199.
48. S. Olariu, An optimal greedy heuristic to color interval graphs, *Inf. Process. Lett.* **37** (1991) 21–25.
49. M. Pal and G. P. Bhattacharjee, The parallel algorithms for determining edge-packing and efficient edge dominating sets in interval graphs, *Parallel Algorithms Appl.* **7** (1995) 193–207.
50. B. S. Panda and D. Pradhan, Minimum paired-dominating set in chordal bipartite graphs and perfect elimination bipartite graphs, *J. Combin. Optim.* **26** (2013) 770–785.

51. A. Pnueli, A. Lempel and S. Even, Transitive orientation of graphs and identification of permutation graphs, *Canad. J. Math.* **23** (1971) 160–175.
52. T. Pramanik, S. Mondal and M. Pal, Minimum 2-tuple dominating set of an interval graphs, *Int. J. Combin.* **2011** (2011) 1–14.
53. H. Qiao, L. Kang, M. Gardei and D. Z. Du, Paired-domination of trees, *J. Global Optim.* **25** (2003) 43–54.
54. A. Rana, A. Pal and M. Pal, An efficient algorithm to solve the distance k-domination problem on permutation graphs, *J. Discrete Math. Sci. Cryptogr.* **19**(2) (2016) 241–255.
55. E. Shan and Y. Dong, The k-tuple twin domination in generalized de Bruijn and Kautz networks, *Comput. Math. Appl.* **63**(1) (2012) 222–227.
56. P. J. Slater, R-domination in graphs, *J. ACM* **23**(2) (1976) 446–450.
57. D. P. Sumner and P. Blitch, Domination critical graphs *J. Combin. Theory Ser. B* **34** (1983) 65–76.
58. R. E. Tarjan, Depth first search and linear graph algorithm, *SIAM J. Comput.* **2** (1972) 146–160.
59. M. Yannakakis and F. Gavril, Edge domination sets in graphs, *SIAM J. Appl. Math. B* **38** (1980) 364–372.