Opuscula Mathematica • Vol. 32 • No. 4 • 2012

http://dx.doi.org/10.7494/OpMath.2012.32.4.707

BOUNDS ON PERFECT k-DOMINATION IN TREES: AN ALGORITHMIC APPROACH

B. Chaluvaraju and K.A. Vidya

Abstract. Let k be a positive integer and $G = (V, E)$ be a graph. A vertex subset D of a graph G is called a perfect k-dominating set of G if every vertex v of G not in D is adjacent to exactly k vertices of D. The minimum cardinality of a perfect k-dominating set of G is the perfect k-domination number $\gamma_{kp}(G)$. In this paper, a sharp bound for $\gamma_{kp}(T)$ is obtained where T is a tree.

Keywords: k-domination, perfect domination, perfect k-domination.

Mathematics Subject Classification: 05C69, 05C70.

1. INTRODUCTION

All graphs considered here are finite, undirected with no loops or multiple edges. Let $n = |V|$ and $m = |E|$ denote the number of vertices and edges of a graph G, respectively. In general, we use $\langle X \rangle$ to denote the sub graph induced by the set of vertices X and $N(v)$ and $N[v]$ denote the open and closed neighborhoods of a vertex v, respectively. Let deg (v) be the degree of vertex v and $\delta(G) = \delta$ the minimum degree and $\Delta(G) = \Delta$ the maximum degree. For graph-theoretical terminology and notation not defined here we follow [9].

A set D of vertices in a graph G is a dominating set if every vertex in $V - D$ is adjacent to some vertex in D. The domination number $\gamma(G)$ is the minimum cardinality of a dominating set of G. A minimum dominating set of a graph G is called a γ-set of G. For a complete review on the topic of domination, see [1, 10] and [11].

For a positive integer k, a vertex subset D of a graph G is called a k-dominating set of G, abbreviated k-DS, if any vertex v of V not in D is adjacent to at least k vertices of D. The minimum cardinality of a k-DS of G is the k-domination number $\gamma_k(G)$ [7]. A vertex subset D of a graph G is called a perfect dominating set of G if any vertex v of G not in D is adjacent to exactly one vertex in D. The concept of perfect domination was introduced by Cockayne, Hartnell, Hedetniemi and Laskar

[3]. For more details on perfect domination and its related parameters, see [2, 5, 6] and [12].

For a positive integer k, a vertex subset D of a graph G is called a perfect k-dominating set of G, abbreviated PkD-set, if any vertex v of G not in D is adjacent to exactly k elements of D. The minimum cardinality of a PkD-set of G is the perfect k-domination number $\gamma_{kp}(G)$ [7]. In fact when $k = 1$, perfect 1-dominating set (P1D) of G is the perfect dominating set of G. Note that every nontrivial graph G has a PkD-set since $V(G)$ is such a set. Also there are graphs whose only perfect k-dominating set is V. A graph for which $\gamma_{kp}(G) < n$ is called a PkD-graph. A tree for which $\gamma_{kp}(T) < n$ is called a PkD-tree. The concept of perfect k-domination was studied by Chaluvaraju, Chellali and Vidya [4].

A possible application to perfect k-domination is a specialist giving radiation (or some powerful drug) to a patient. In order to be effective there must be precisely k units administered to the neighboring cells (any more is very dangerous). The cells where the drug is directly given are, unfortunately, weakened or harmed and to minimize the number of damaged cells we wish to minimize the number of spots/cells where the drug is given. Thus we would want a minimum perfect k-dominating set.

In this paper we have studied perfect k-domination in trees as a special case and we have developed an algorithm to find the k-domination number of a tree and another algorithm to check whether a tree is a PkD-tree or not. Extending this algorithm we can also find a minimal PkD-set for a tree. Using these algorithms we establish a sharp bound for $\gamma_{kp}(T)$.

We recall that $\alpha_0(G)$ is the vertex covering number and $\beta_0(G)$ is the independence number of a graph G.

Observation 1.1. *For every bipartite graph $\beta_0(G) \geq \lceil n/2 \rceil$ and therefore for every tree $\beta_0(G) \geq \lceil n/2 \rceil > (n-1)/2$. Similarly $\alpha_0(G) > (n-1)/2$.*

Theorem 1.2 ([9])**.** *For any graph G,*

$$\alpha_0(G) + \beta_0(G) = n = \alpha_1(G) + \beta_1(G).$$

Theorem 1.3 ([4])**.** *For any non trivial graph G,*

$$n - (m/k) \leq \gamma_{kp}(G) \leq n.$$

Theorem 1.4. *For any PkD-tree T:*

(i) $\gamma_{kp}(T) \geq \alpha_0(T)$,
(ii) $\gamma_{kp}(T) \geq \beta_1(T)$.

Proof. By Observation 1.1, we have $n - ((n-1)/k) \leq \gamma_{kp}(T)$, since $m = n - 1$. Therefore using Theorem 1.2 and Theorem 1.3,

$$\gamma_{kp}(T) \geq n - ((n-1)/k) \geq n - ((n-1)/2) \geq n - \beta_0(T) \geq \alpha_0(T).$$

Similarly (ii) follows. □

Theorem 1.5. *Let T be a PkD-tree. Then at least one of the following holds:*

(i) *There exists at least one vertex of degree k in T.*
(ii) *There exists at least two vertices u and v in T of degree $k+1$ such that every vertex in the u-v path has degree greater than $k+1$.*

Proof. Let D be a PkD-set of a PkD-tree T. Suppose (i) does not hold. Then we have to prove that (ii) holds. Hence we consider the following cases.

Case 1. There exist at least one vertex of degree $k+1$ in $V-D$.
Let u be a vertex of degree $k+1$ in $V-D$. Since $u \in V-D$ and $deg(u) = k+1$, u is adjacent to a vertex $u_1 \in V-D$. Again since $u \in V-D$, $deg(u_1) \geq k+1$. If $deg(u_1) = k+1$, (ii) holds. If not, u_1 is adjacent to at least one vertex $u_2 \in V-D$. Again continuing the same arguments we can conclude that (ii) holds, since T is a finite graph.

Case 2. $V-D$ does not have a vertex of degree $k+1$.
Let u be a vertex in $V-D$ of degree greater than $k+1$. Then u has at least 2-neighbors in $V-D$. Let u_1 be one such neighbor. Since $u_1 \in V-D$ and $deg(u_1) > k+1$, u_1 has a neighbor $u_2 \in V-D$ and $deg(u_2) > k+1$. This argument never ends and so contradicts the fact that T is finite. Thus the only possibility, is case 1 for which (ii) holds. □

2. CHARACTERIZATIONS OF GRAPHS WITH DISJOINT PkD-SETS

In the previous paper [4] we obtained some bounds on a perfect k-domination number. Now naturally comes the interesting question, when will a graph have two disjoint perfect k-dominating sets? Though we couldn't characterize all the graphs with this property, we found some observations related to this question.

Observation 2.1. *A graph will have two disjoint perfect k-dominating sets only if $k \leq \delta$, since all the vertices with degree less than k belongs to every PkD-set. Thus a tree will not have more than one disjoint perfect k-dominating set.*

Lemma 2.2. *If G has two disjoint PkD-sets D_1 and D_2, then they are of the same cardinality.*

Proof. If possible let D_1 and D_2 be two disjoint PkD-sets with $|D_1| = r$ and $|D_2| = s$, $r \neq s$. Then to satisfy the perfectness condition of vertices in D_1, there should be ks-edges from D_1 to D_2. Similarly to satisfy the perfectness condition of D_2 there should be kr-edges from D_2 to D_1. This implies $kr = ks$, which is a contradiction since $r \neq s$. Thus two disjoint PkD sets will have the same cardinality. □

Theorem 2.3. *In a graph G, if there exist a PkD-set which is disjoint from all other PkD sets, then all the PkD-sets of G have the same cardinality.*

Theorem 2.4. *If G has two disjoint PkD-sets D_1 and D_2 for $k > \Delta/2$, then $\gamma_{kp}(G) = n/2$.*

Proof. Let D_1 and D_2 be two disjoint PkD-sets. Suppose $\gamma_{kp}(G) < n/2$. Then there exists at least one vertex which does not belong to both the PkD-sets. Then that vertex is adjacent to k vertices in D_1 as well as k vertices in D_2. Therefore $\Delta \geq 2k$. Thus $k < \Delta/2$. Hence if G has two disjoint PkD-sets for $k > \Delta/2$, then $\gamma_{kp}(G) = n/2$. $\square$

Corollary 2.5. *A graph with odd number of vertices does not have two disjoint PkD-sets for $k > \Delta/2$.*

Theorem 2.6 ([4])**.** *The perfect k-dominating set is NP-complete.*

Though the problem of finding a perfect k-dominating set is NP-Complete for general graphs, it might be possible to find polynomial time algorithms for finding the perfect k-domination number in some subclasses. As an initiative to this study, we take the subclass of trees, and study the perfect k-domination in trees.

To find a lower bound for $\gamma_{kp}(T)$, we developed the following algorithm. From the definitions we know $\gamma_k(T) \leq \gamma_{kp}(T)$. Given a tree and a number k, the algorithm below gives the k-domination number of the tree. For this algorithm, first we have to assign numbers to the vertices in such a way that the tree will be a rooted tree at vertex v_1 and the assigned number to the vertices will increase as we go to the upper branches of the tree in a Breadth First Search fashion. For more details on NP-completeness see [8].

Algorithm to find k-domination number of a tree:

Step 1. Assign a label $Lv[i] = N$ to every vertex. ($'N'$ stands for $'$not sure to be in D' and $'S'$ stands for $'$sure to be in D', where D is the PkD-set.)

Step 2. Assign a number Dv for each vertex, where $Dv[i]$ denotes number of vertices adjacent to v_i with $Lv[i] = S$, excluding its parent vertex. First we set the value of $Dv[i]$ to 0 for every i and using the following procedure, we will find exact values of $Dv[i]$ for each vertex.

Step 3. Starting from the top numbered vertex, we will check Dv of each vertex. For a vertex v_i with $Lv[i] = 'N'$:

(i) If $Dv[i] = k - 1$, we can take that vertex out from D to $V - D$ and change the label of its parent vertex to $'S'$.
(ii) If $Dv[i] < k - 1$, then that vertex v_i has to be included in D. So change the label $Lv[i] = 'N'$ to $'S'$ and we will increase the number Dv of its parent vertex by 1. Continue the process with the next vertex v_{i-1}.
(iii) If $Dv[i] \geq k$, we include that vertex in $V - D$. Continue the process with next numbered vertex (it may not be its parent vertex).

For the vertex v_i with $Lv[i]='S'$, increase the number Dv of its parent vertex by 1. Continue the process with the next vertex v_{i-1}.

Step 4. Follow these steps till we reach the last vertex v_1. Since it does not have a parent vertex, necessary changes in conditions of step 3 are done:

(i) If $Dv[1] \leq k-1$, then v_1 has to be included in D. So change the label $Lv[1] = 'N'$ to $'S'$.
(ii) If $Dv[1] \geq k$, we include that vertex in $V - D$.

Step 5. All the vertices with label $'S'$ form a k-dominating set of G. The number of vertices with label $'S'$ gives the k-domination number of the tree.

From the algorithm we get the k-domination number of the tree. Let $a = \gamma_k(T)$. Then clearly $a \leq \gamma_{kp}(T)$.

Correctness of the algorithm:

Case 1. Let u be a vertex with $Dv[i] < k-1$. Then u belongs to all the k-dominating sets of G. This justifies the inclusion of u in D.

Case 2. Let u=v_i be the first vertex for which $Dv[i] \geq k-1$. In the algorithm, we have included u in $V - D$. Then let $u \notin D$ where D is a k-dominating set of G. Now we have to check whether there exists a minimum dominating set D_1 with $|D_1| < |D|$ and $u \in D_1$. If $u \in D_1$, then u is included in D_1, only for taking its parent vertex v, out of D_1. But then the k-domination number is not reduced, since u is added to D_1, though v is taken out of it. Other vertices are included in D or $V - D$ irrespective of the status of u. This proves the correctness of the algorithm by taking $u \notin D$. Continuing this process we get the k-domination number of the tree.

As mentioned earlier there are graphs whose only perfect k-dominating set is V. A tree for which $\gamma_{kp}(T) \neq n$ is called a PkD-tree. So given a tree and k, first we have to find whether the tree is a PkD-tree or not. The algorithm below finds whether a tree is a PkD-tree or not.

Algorithm to check whether the given tree is PkD tree or not:

Step 1. Given a tree assign numbers to the vertices in such a way that the tree will be a rooted tree at vertex v_1 and the assigned number to the vertices will increase as we go to the upper branches of the tree in a breadth first search fashion. (As in previous algorithm)

Step 2. Assign a label $Lv[i] = S$ to every vertex. ($'N'$ stands for $'$not sure to be in D' and $'S'$ stands for $'$sure to be in D', where D is the PkD-set.) Now we will try to remove at least one vertex from D to $V - D$, so that T is a PkD-tree.

Step 3. Assign two numbers Dv and Nv for each vertex, where $Dv[i]$ denotes number of vertices adjacent to v_i, which are $'$sure to be in D', excluding its parent vertex. Similarly $Nv[i]$ denotes number of vertices adjacent to v_i, which are $'$not sure to be in D', excluding its parent vertex. First we set those values to 0 and using the following procedure, we will find exact values of $Dv[i]$ and $Nv[i]$ for each vertex.

Step 4. Starting from the top numbered vertex, we will check Dv and Nv of each vertex. For a vertex v_i:

(i) If $Dv[i]$ is $k-1$, we can take that vertex out from D to $V-D$ since v_i is adjacent to k vertices with label $'S'$ (including its parent vertex) and so T is a PkD-tree. So change $Lv[i] = N$.
(ii) If $Dv[i]$ is less than $k-1$ or greater than k, that vertex has to be in D and so we will increase Dv of its parent vertex by 1 and continue the process with next vertex v_{i-1}.
(iii) If $Dv[i]$ is equal to k, we can add that vertex in $V-D$ provided its parent vertex is also in $V-D$. In that case, we change labels of both vertices to $'N'$ and continue the process with the next numbered vertex. Let v_m be the parent vertex of the vertex v_i.

Case 1. If $Dv[m] = k-1$, then we can include v_i, v_m and all other vertices with label $'N'$ in $V-D$. Then T is a PkD-tree.

Case 2. If $Dv[m] < k-1$, check $Nv[m] > k-1-Dv[m]$. If yes, from $Nv[m]$ neighbors of v_m with label $'N'$, change the label of $k-1-Dv[m]$ neighbors to $'S'$. Then we can include v_i, v_m and all other vertices with label $'N'$ in $V-D$. Then T is a PkD-tree. If $Nv[m] < k-1-Dv[m]$, change the label of v_m and label of all the vertices in its branches to $'S'$ and continue as in step 4 (ii).

Case 3. If $Dv[m] > k$ change the label of v_m and label all of the vertices in its branches to $'S'$ and continue as in step 4 (ii).

Case 4. If $Dv[m] = k$, we have to continue the same process as in step 4 (iii).

Step 5. At any stage if we get a result T is a PkD tree, the process is stopped. Let T' be the connected induced subgraph of T whose perfect k-domination number is less than $|T'|$. Otherwise follow these steps till we reach the last vertex v_1. Since it does not have a parent vertex, necessary changes in conditions of step 4 are done. Thus we can check whether the given tree is a PkD tree or not.

The algorithm can be extended to find a minimal perfect k-dominating set of T. Even after getting the result that T is a PkD-tree before reaching v_1, if we continue following this algorithm till we reach v_1, after fixing the labels of the connected induced subgraph T' of T whose perfect k-domination number is less than $|T'|$, we get a minimal perfect k-dominating set of T. Let $'d'$ be the cardinality of this minimal perfect k-dominating set. Since minimal perfect k-domination number is greater than or equal to the minimum perfect k-domination number, we have $\gamma_{kp}(T) \leq d$.

Correctness of the algorithm:

Here we modify the algorithm for k-domination to include the perfectness condition. But deciding whether a vertex u is to be included or not to be included in D, is difficult. So instead of finding minimum k-perfect dominating set, we try to find a connected induced subgraph T' of T whose perfect k-domination number is less than $|T'|$. We continue this process till the last vertex and the resulting k-perfect dominating set will be a minimal k-perfect dominating set, but need not be the minimum.

Thus using the above two algorithms we get a bound for $\gamma_{kp}(T)$, that is given a tree and a number k we get two numbers 'a' and 'd' such that a $\leq \gamma_{kp}(T) \leq d$, which gives an upper and lower bound for the problem. Also the bounds are sharp as we can see from the following example. Consider the star $K_{1,t}$. Let $k = t$. Then $\gamma_k(T) = t$ and so $a = t$, and also $d = t$. Therefore the bounds are sharp.

Complexity of the algorithm:

After assigning numbers to the vertices of the tree, each vertex is checked only once to find the Dv of the vertex. So complexity of the algorithm is of $O(n)$.

Acknowledgements
We are grateful to the referees for their careful reading with corrections and useful comments.

REFERENCES

[1] B.D. Acharya, H.B. Walikar, E. Sampathkumar, *Recent developments in the theory of domination in graphs*, Mehta Research Institute, Allahabad, MRI Lecture Notes in Math., 1979.

[2] D. W. Bange, A.E. Barkauskas, P.J. Slater, *Efficient dominating sets in graphs*, [in:] R.D. Ringeisen, F.S. Roberts, eds, Applications of Discrete Mathematics (SIAM, Philadelphia, 1988), 189–199.

[3] E.J. Cockayne, B.L. Hartnell, S.T. Hedetniemi, R. Laskar, *Perfect domination in graphs*, J. Combin. Inform. System Sci. **18** (1993), 136–148.

[4] B. Chaluvaraju, M. Chellali, K.A. Vidya, *k-Perfect domination in graphs*, Australasian Journal of Combinatorics **48** (2010), 175–184.

[5] I.J. Dejter, J. Pujol, *Perfect Domination and Symmetry*, Congr. Numer. **111** (1995) 18–32.

[6] M.R. Fellows, M.N. Hoover, *Perfect domination.* Australasian Journal of Combinatorics **3** (1991), 141–150.

[7] J.F. Fink, M.S. Jacobson, *n-domination in graphs*, [in:] Y. Alavi and A. J. Schwenk, eds, Graph Theory with Applications to Algorithms and Computer Science, Wiley, NewYork, 1985, 283–300.

[8] M.R. Garey, D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-completeness.* W.H. Freeman, San Francisco, 1979.

[9] F. Harary, *Graph theory*, Addison-Wesley, Reading, Mass., 1969.

[10] T.W. Haynes, S.T. Hedetniemi, P.J. Slater, *Fundamentals of domination in graphs*, Marcel Dekker, Inc., New York, 1998.

[11] T.W. Haynes, S.T. Hedetniemi, P.J. Slater, *Domination in graphs: Advanced topics*, Marcel Dekker, Inc., New York, 1998.

[12] M. Livingston, Q.F. Stout, *Perfect dominating sets*, Congr. Numer. **79** (1990), 187–203.

B. Chaluvaraju
bchaluvaraju@yahoo.co.in

Department of Mathematics
Bangalore University
Central College Campus
Bangalore -560 001, India

K.A. Vidya
vidya.mnj@gmail.com

Department of Mathematics
Bangalore University
Central College Campus
Bangalore -560 001, India

Received: September 27, 2010.
Revised: January 14, 2012.
Accepted: April 3, 2012.