

# Algorithmic Aspects for Total Connected Dominating Sets in Mobile Ad Hoc Wireless Networks

Mr. C.D.Guruprakash<sup>1</sup>, Dr.B.P.Mallikarjunaswamy<sup>2</sup>

Department of Computer Science and Engineering, Sri Siddhartha Institute of Technology

Maralur, Tumkur-572105, Karnataka

E-mail:cdguruprakash@gmail.com,drbpmwamy@rediffmail.com.

**Abstract-** A Total connected dominating set (TCDS) for a graph  $G(V, E)$  is a subset  $V'$  of  $V$ , such that each node in  $V - V'$  is adjacent to some node in  $V'$  and  $V'$  induces a connected subgraph.

A TCDS has been proposed as a virtual backbone for routing in wireless ad hoc networks. However, it is NP-hard to find a minimum connected dominating set (MCDS). Approximation algorithms for MCDS have been proposed in the literature. Most of these algorithms suffer from a very poor approximation ratio, and from high time complexity and message complexity. Recently, new distributed heuristics for constructing a TCDS were developed, with constant approximation ratio of 8. These new heuristics are based on a construction of a spanning tree, which makes it very costly in terms of communication overhead to maintain the TCDS in the case of mobility and topology changes.

Investigating the algorithmic complexity of total domination for wireless ad hoc network as relevant open question. An  $O(n^2)$  algorithm has been proposed for this problem by Bertossi [1]. Keil [10] proposed an  $O(n + m)$

In this paper, we propose the first distributed approximation algorithm to construct a MCDS for the unit-disk-graph with a constant approximation ratio  $O(n)$ , and linear time and linear message complexity. This algorithm is fully localized, and does not depend on the spanning tree. Thus, the maintenance of the TCDS after changes of topology guarantees the maintenance of the same approximation ratio. In this algorithm each node requires knowledge of its single hop neighbors, and only a constant number of two-hop and three-hop neighbors.

**Index Terms-** Total Connected Dominating Set, Independent Set, ad hoc networks, connected dominating set, maximal independent set, mobility.

## 1. Introduction

In wireless ad hoc networks, there is no fixed or pre-defined infrastructure. Nodes in wireless networks communicate via a shared medium, either through a single hop or multihops. Although there is no physical backbone infrastructure, a virtual backbone can be formed by constructing a Total Connected Dominating Set (TCDS). Given an undirected graph  $G = (V; E)$ . A matching in  $G$  is a subset  $M$  of edges of  $E$  such that no two edges in  $M$  are adjacent. A matching  $M$  in  $G$  is called a perfect matching if every vertex of  $G$  is incident to some edge in  $M$ .

A dominating set  $D$  of  $G$  is said to be a matching dominating set if the induced sub graph  $\langle D \rangle$  admits a perfect matching. The cardinality of the smallest matching dominating set is called matching domination number.  $V' \subseteq V$  is a TCDS of  $G$  if for each node  $u \in V - V'$   $u$  is either in  $V'$  or there exists a node  $v \in V'$  such that  $uv \in E$  and the sub graph induced by  $V'$ , i.e.,  $G(V')$ , is connected. The nodes in the TCDS are called dominators, other nodes are called dominatees. With the help of the TCDS, routing is easier and can adapt quickly to network topology changes. To reduce the traffic during communication and simplify the connectivity management, it is desirable to construct a Total connected dominating set.

The CDS problem has been studied intensively in Unit Disk Graph (UDG), in which each node has the same transmission range. The TCDS problem in UDG has been shown to be NP-complete [6]. To build a CDS, most of current algorithms first find a Maximal Independent Set (MIS)  $I$  of  $G$  and then connect all nodes in  $I$  to have a CDS. The independent set  $I$  is a subset of  $V$  such that for any two nodes  $u, v \in I, uv \notin E$ . In other words, the nodes in  $I$  are pair wise nonadjacent. A maximal independent set is an independent set such that no more nodes can be added to remain the non adjacency property. The most relevant related work

using this scheme are in [2], [3]. In [2], Wan et al. proposed the first distributed algorithm with the performance ratio of 8. Later, Li et al. proposed a better algorithm with the performance ratio of  $(4:8 + \ln 5)$  by constructing a Steiner tree when connecting all nodes in  $I$  [3]. However, the algorithmic complexity of total domination in mobile ad hoc network, we are presenting an  $O(n)$  time algorithm for finding the total set of a circular networks.

## 2. RELATED WORK

Since interference is a major factor limiting performance in wireless multi-hop networks, a lot of effort has been invested in deriving realistic interference models, and to analyze network performance under such models. A seminal

work in this area is [12], in which Gupta and Kumar study the transport capacity of wireless networks under two different interference models, the physical and the protocol interference model. Contrary to the physical model, which is used in this paper, the protocol interference model is a localized model, since decision on whether a certain communication is successful depends only on the presence of concurrent transmitters within a bounded area centered at the receiver.

Another localized interference model commonly used in the

literature is the graph-based interference model, in which a certain communication graph representing communication links is assumed, and only links whose endpoints are up to a certain hop distance  $d$  on the communication graph from link  $(u, v)$  can interfere with  $(u, v)$  [29].

Due to their simplicity and the fact that they somehow resemble the behavior of the 802.11 MAC layer, localized interference models have been mostly used in the literature to design interference-aware protocols. This is the case, for instance, for the protocols presented in [2, 21, 25, 26]. Given the complexity of dealing with physical interference, only a few protocols based on this model have been proposed so far. For example, [10, 11, 14] consider the problem of scheduling transmissions, but they provide solutions which are computationally infeasible even for a small size network. Only recently, a computationally efficient algorithm for scheduling transmissions under the physical interference model with a provable approximation bound has been proposed [4]. The physical interference model has been recently used in [9, 23] to study the complexity of scheduling a set of link demands in the shortest possible time and of one-shot scheduling (scheduling as many transmissions as possible in a single communication slot), which are both shown to be NP complete for wireless networks in a 2-dimensional Euclidean space. Various distributed algorithms have been proposed for finding good approximations of minimum dominating sets in arbitrary graphs (see, for example, [8, 19, 20, 22]). Alzoubiet et al. [3] presented the first

constant approximation algorithm for the minimum connected dominating set problem in unit-disk graphs with  $O(n)$  and  $O(n \log n)$  time and message complexity, respectively. Cheng et al. [5] proposed a polynomial time approximation scheme for the connected dominating set problem in unit-disk graphs. Huang et al. [13] formally analyze a popular algorithm used for clustering in ad-hoc mobile network scenarios. They show that this algorithm gives a 7-approximation for the minimum dominating set problem in unit-disk graphs while being able to adapt to the mobility of the nodes in the network.

Kuhn et al. [19] presented a distributed algorithm that computes a constant factor approximation of a minimum dominating set in  $O(\log^2 n)$  time without needing any synchronization but it requires that nodes know an estimate of the total number of nodes in the network. In [24], Parthasarathy and Gandhi also present distributed algorithms to compute a constant factor approximation to the minimum dominating set. The running time of their algorithm depends on the amount of information available to the nodes, and nodes have to know an estimate of the size of the network. Both papers extend the unit-disk model taking into account signal interference. A more realistic model taking physical carrier sensing into account was considered by Kothapalli et al. [16], but their algorithm needs  $O(\log^4 n)$  time steps, w.h.p., in order to construct a constant factor approximation of a minimum dominating set.

## 3. PRELIMINARIES

Generally speaking, a communication network is modeled as an undirected graph  $G = (V, E)$ , where the vertices  $V$  represent the  $n := |V|$  devices or nodes, and two nodes are connected if they can communicate directly with one another. Consequently, a wireless network is created by the communication links between a collection of radio transceivers. Two vertices of a graph are called independent if they are not adjacent to one another. A subset  $I \subseteq V$  is called independent if all vertices are not connected. A subset  $D \subseteq V$  is called dominating if every vertex from  $V$  is contained in this subset, or adjacent to a vertex from  $D$ . A subset can be both independent and dominating. Such a set is also called maximal independent set. Formally, an independent set is called maximal if it cannot be extended by the addition of any other vertex from the graph without violating the independence property. It is easy to verify that a maximal independent set is dominating property. It is easy to verify that a maximal independent set is dominating. If we seek for a small subset of vertices that is both dominating and independent, we call the resulting problem Minimum Independent Dominating Set problem (Min-IDS). Since any maximal independent set also dominates a graph, the Min-IDS problem is sometimes also referred to as Minimum Maximal Independent Set problem.

Consider a circular arc network  $G(F)$ . Without loss of generality, assume that all arc end points are distinct and no arc covers the entire circle. If an arc  $i$  begins at the point  $p$  and ends at the point  $q$  In the clock wise direction then define  $p$  to be the head of  $i$  (denoted by  $h(i)$ ) and  $q$  to be the tail (denoted by  $t(i)$ ). The continuous part of the circle that begins at  $p$  and ends at  $q$  in the clockwise direction is referred to as segment  $(p, q)$  of the circle.

$(p, q)$  is used to denote the segment  $(p, q)$  including the points  $p$  and  $q$ . Define  $(p, q)$  to be the segment containing  $p$  but not  $q$ . Similarly  $(p, q)$  is defined. Starting with an arbitrary arc, label the  $n$  arcs from 1 through  $n$  in ascending order of their tail ends. We use the terms arc in  $F$  and vertex in  $G$  to mean the same.

For an arc  $i$  in  $F$ , define  $N(i, F)$  as the set of all arcs in  $F$  that intersect with  $i$ . For an arc  $i$  in  $F$ , define  $DOMARC(i)$  to be the arc  $J$ . In  $F$  if  $J$  is in  $N(i, F)$  and  $t(J)$  is encountered last among all tails of arcs in  $N(i, F)$  in a clock wise traversal from  $t(i)$  including  $t(i)$ . The arc  $DOMARC(i)$  will dominate  $i$  as well all as maximum number of arcs in a clock wise traversal from  $t(i)$ .

For any arc  $i$  in  $F$ , the first undominated arc  $FUD(i)$  is  $J$  if  $J$  belongs to  $F \setminus N(i, F)$  and  $t(J)$  is the first tail encountered on a clockwise traversal from  $t(i)$ . If  $i$  is included in the total dominating set then  $FUD(i)$  will be the first undominated arc after  $i$  on clockwise traversal from  $t(i)$ .

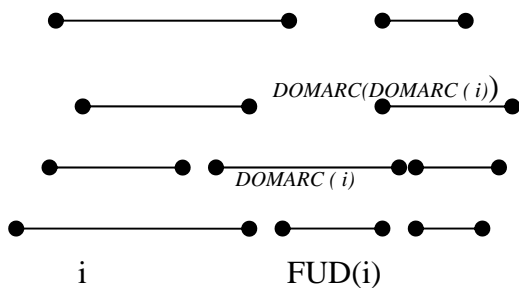


Fig.1 Illustration of  $DOMARC(i)$

#### 4. ALGORITHM

In this section, we present the algorithm that yields a to construct a dominating set.

---

Algorithm : MTDS.

---

1. For each node  $i$  in  $F$ , set  $FLAG(i) = 0$
  2. Let  $x$  be any arbitrary node
  3.  $Prev = nil; i = FUD(x);$   
 $leadarc = DOMARC(i)$
  4. Repeat step 5 to 7 while  $FLAG(prev) = 0$
  5.  $j = DOMARC(i)$
  6.  $k = DOMARC(DOMARC(i))$
  7. If  $j$  belongs to  $N(prev, F)$ , then  
 $prev = j; i = FUD(j)$   
Otherwise  
If  $j = k$  \* $j$  is an arc\*  
 $prev = DOMARC(prev); i = FUD(prev)$   
Otherwise  
 $prev = k; i = FUD(k); FLAG(j) = 1;$   
 $leadrac = j$
  8. Find  $MTDS(leadarc)$ .
  9. End.
- 

Step 1 is used to set a flag as zero for each arc in  $F$ . The process can start from any arc  $x$ . For the arcs which are identified as leading arcs, the flag is set as 1 whenever that arc is visited. This flag is used to identify the completion of the cycle in the directed graph. Step 4 will identify the completion of the cycle. At step 5 and step 6, we identify a pair of arcs which intersects each other so that they will dominate each other if they are included in the total dominating set. Among the two arcs  $j$  and  $k$  identified, step 7 will check whether  $j$  is dominated by the previous arc. In that case only  $j$  is included in the cycle and a better arc than  $k$  is identified by  $DOMARC(FUD(j))$ . At the end of this procedure we identify an arc as leadarc. The next procedure MTDS starts with this arc.

Among the arcs in the given graph, the algorithm identifies some arcs as leading arcs. Each leading arc  $i$

will have a follower given by  $DOMARC(i)$  so that if  $i$  is included in the total dominating set the  $DOMARC(i)$  is also likely to be included in the total dominating set. These arcs are dominating each other; if both these arcs are included in the total dominating set each follower arc may have followers for it. Whether a follower arc  $i$  has its follower is decided by  $DOMARC(FUD(i))$ . If  $DOMARC(FUD(i))$  overlaps with  $i$ , then  $DOMARC(FUD(i))$  becomes a follower if  $i$ . Otherwise  $DOMARC(FUD(i))$  will be designated as the next leading arc.

**5.Results and Discussion.**

Consider the graph given Fig.2 for each node  $FUD(i)$ ,  $DOMARC(i)$ , and  $DOMARC(DOMARC(i))$  are given Table 1.

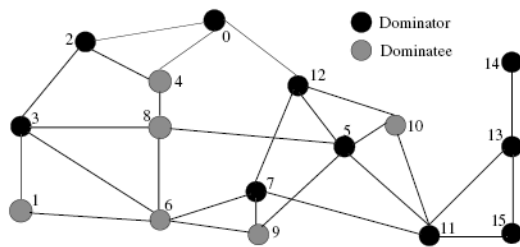


Fig.2 Model of the graph.

If we start with node 1, then the sequence of subsets generated by the above algorithm is (5,6), (10,11), (2,5), (9,10), (2,5). Here the highlighted nodes 5, 10, 2 and 9 are the leading nodes. Node 6 is the follower of node 5 and node 11 is the follower of node 10. By executing the above algorithm starting with node 1, we can see that the node 2 is repeated as leading node and hence the next algorithm MTDS is executed starting with node 2.

Table.1 Computed values for the graph given fig.2

Node	FUD(i)	DOMNODE(i)	DOMNODE(DOMNODE(i))
1	4	5	6
2	4	5	6
3	6	5	6
4	6	5	6
5	7	6	8
6	9	8	9
7	10	9	10
8	10	9	10
9	11	10	11
10	12	11	12
11	13	12	2
12	14	2	5
13	1	2	5
14	3	2	5

Through the algorithm, we construct a sequence of subsets of nodes, with each subset containing a minimum of two nodes. The algorithm sets the flag as 1 for the first element in each subset identified as the leading arcs. Let N denotes the set of all leading nodes.

Algorithm 2: MTDS(i)

1. For each node  $i$  in  $F$ , set  $FLAG(i) = 0$
2. Let  $X$  be any arbitrary arc
3.  $prev = nil$ ;  
 $i = FUD(x); leadarc = DOMARC(i)$
4. Repeat steps 5 to 7 while  $FLAG(prev) = 0$
5.  $j = DOMARC(i)$
6.  $k = DOMARC(DOMARC(i))$
7. If  $j$  belongs to  $N(prev, F)$  then  
 $prev = j; i = FUD(j)$   
Otherwise  
If  $j = k \setminus j$  is an end arc
- $prev = DOMARC(prev); i = FUD(prev)$   
Otherwise  
 $prev = k; i = FUD(k); FLAG(j) = 1$ ;  
 $leadarc = j$ .
8. Find MTDS(leadarc)
9. end

**Lemma 1.1.** MTDS is a dominating set. We show that every node  $i$  in  $F$  intersect with some node in MTDS. The algorithm finds MTDS as  $S_1, S_2, \dots, S_m$  where  $S_i$  is

a set of nodes which intersects with at least with one node in the same set. Let  $h_i$  indicate the head of the first node in  $S_i$  and  $t_i$  indicate the tail of the last node in  $S_i$ . Consider an node  $i$  in  $F$ . If  $t(i)$  falls in the segment  $(h_k, t_k)$  for some  $K$ , then clearly some node in  $S_k$  dominates  $i$ . If  $t(i)$  does not fall in  $(h_k, t_k)$  for any  $K$ , then let  $x$  be the node in MTDS before  $i$  and  $y$  be the node in MTDS after  $i$ . Then by definition,  $y = \text{DOMNODE}(FUD(x))$  and as per the assumption,  $i$  does not intersect with  $x$ . Now two cases are possible

**Case 1.**  $i = FUD(x)$ . In this case  $y = \text{DOMNODE}(i)$

And  $i$  is dominated by  $y$ , a contradiction.

**Case 2.**  $i \neq FUD(x)$ . In this case  $t(i)$  lies in  $t(FUD(x), t(y))$  and  $h(y)$  lies in  $(t(x), t(FUD(x)))$ .

Then  $i$  has to intersect with  $y$  and hence is dominated by  $y$ ,

a contradiction. Hence every node in  $F$  is dominated by some node in MTDS.

**Lemma 1.2.** MTDS is a total dominating set.

To show that MTDS is total, it is necessary to show that every node in MTDS intersect with some other node in MTDS.

Let  $i$  be an node in MTDS. Let  $J$  be the node in MTDS just before  $i$ . As per the algorithm, three cases can arise.

**Case 1.**  $i = \text{DOMNODE}(j)$ , then clearly  $i$  intersect with  $j$  and hence dominated by  $J$ .

**Case 2.**  $i = \text{DOMNODE}(FUD(j))$  and  $i$  intersect with  $j$ .

**Case 3.**  $i = \text{DOMNODE}(FUD(j))$  and  $i$  is not adjacent to  $j$ . Then  $K = \text{DOMNODE}(i)$  is in MTDS and is adjacent with  $i$ . Thus, every node in MTDS intersect with some other node in MTDS and hence MTDS is a total dominating set.

**Lemma 1.3.** MTDS is minimum.

Let  $MTDS = \{S_1, S_2 \dots S_m\}$  where each  $S_i$  is a set of node which intersect with at least one node in the same set. Let  $l_i$  be the last node in  $S_i$ . Then the entire network can be partitioned in to  $m$  segments as  $((t(l_1), t(l_2)), (t(l_2), t(l_3)), \dots (t(l_m), t(l_1)))$ .

To prove that MTDS is minimum, we prove that each segment

$((t(l_1), t(l_2)), (t(l_2), t(l_3)), \dots (t(l_m), t(l_1)))$  has minimum number of nodes included in MTDS. We prove that, for any  $i$ , if there are  $n$  nodes in MTDS from set  $S_i$ , there does not exist a total dominating set with  $n - 1$  nodes for this segment. There should be minimum two nodes from each of these segments to have a total dominating set. Hence we need to consider the segments where there are more than two nodes included in MTDS. If there are  $n$  nodes in set  $S_i$  as per the algorithm, there is an node in segment corresponding to set  $S_i$  which is not dominated by the first  $n - 1$  nodes. Hence the minimum number of nodes to be included in MTDS from segment corresponding to set  $S_i$  is  $n$ .

## 6. Conclusion

In this paper, we have studied the total Connected Dominating Set (TCDS) problem in Circular Graphs with only Bidirectional links (CGB). The circular graphs can be used to model wireless ad hoc networks where nodes have different transmission ranges. We have proposed three approximation algorithms and shown that the obtained CDS is within a constant factor of the optimal CDS. The main approach in our algorithms is to construct a maximal independent set and then connect them. Through the theoretical analysis and simulation results, we have shown that using a Steiner tree with the minimum number of Steiner nodes to interconnect the maximal independent set can help to reduce the size of the TCDS. In addition, choosing a node with the largest transmission range as a dominator can further reduce the TCDS size. Moreover, we have also presented the size relationship between an independent set and a TCDS of a given network. We have pointed out some important characteristics of a CGB. In particular, given a CGB, there exists a node such that the maximum number of its independent neighbors is 5. In addition, we have also proved the upper bound of the maximum number of independent neighbors of any node in a CGB. When nodes in a network have different transmission ranges, a node  $u$  can communicate directly to a node  $v$  but node  $v$  might not be able to communicate directly back to node  $u$ . In this case, the edge  $(u; v)$  is a directed edge, called unidirectional links. Thus we are interested to study the CDS problem in the general disk graphs, where both unidirectional and bidirectional links exist. One simple way is to find a dominating set and then use a directed Steiner nodes algorithm to connect them. Note that the CDS in this case is directed. Hence we need to find a strongly connected CDS to help the routine

## References

- [1.] B. Clack, C. Colbourn, and D. Johnson, "Unit Disk Graphs," *Discrete Mathematics*, vol. 86, pp. 165–177, 1990.
- [2.] P.-J. Wan, K. M. Alzoubi, and O. Frieder, "Distributed Construction on Connected Dominating Set in Wireless Ad Hoc Networks", *Proceedings of the Conference of the IEEE Communications Society (INFOCOM)*, 2002.
- [3.] Y. Li, M. T. Thai, F. Wang, C.-W. Yi, P.-J. Wang, and D.-Z. Du, "On Greedy Construction of Connected Dominating Sets in Wireless Networks", *Special issue of Wireless Communications and Mobile Computing (WCMC)*, 2005.
- [4.] S. Funke, A. Kesselman, U. Meyer, and M. Segal, "A Simple Improved Distributed Algorithm for Minimum CDS in Unit Disk Graphs", *1st IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2005.
- [5.] M. Cardei, M.X. Cheng, X. Cheng, and D.-Z. Du, "Connected Domination in Ad Hoc Wireless Networks", *Proceedings of the Sixth International Conference on Computer Science and Informatics (CSI)*, 2002.
- [6.] S. Guha and S. Khuller, "Approximation Algorithms for Connected Dominating Sets", *Algorithmica*, vol. 20, pp. 374–387, 1998.
- [7.] L. Ruan, H. Du, X. Jia, W. Wu, Y. Li, and L.-I. Ko, "A Greedy Approximation for Minimum Connected Dominating Sets", *Theoretical Computer Science*, 2005.
- [8.] J. Wu and H. Li, "On Calculating Connected Dominating Sets for Efficient Routing in Ad Hoc Wireless Networks", *Proceedings of the Third International Workshop Discrete Algorithms and Methods for Mobile Computing and Comm.*, 1999.
- [9.] Y. Li, S. Zhu, M. T. Thai, and D.-Z. Du, "Localized Construction of Connected Dominating Set in Wireless Networks", *NSF International Workshop on Theoretical Aspects of Wireless Ad Hoc, Sensor and Peer-to-Peer Networks*, 2004.
- [10.] T.W. Haynes and P.J. Slater "Paired Domination in Graphs Networks, 32(1998), 199-206.
- [11.] T.W. Haynes and P.J. Slater "Paired Domination and paired dominating number, 209(1995), 65-72.
- [12.] K.M. Alzoubi, P.-J. Wang, and O. Frieder, "Message-Optimal Connected Dominating Sets in Mobile Ad Hoc Networks", *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC)*, 2002.
- [13.] B. Das, R. Sivakumar, and V. Bharghavan, "Routing in Ad Hoc Networks Using a Spine", *International Conference on Computers and Communication Networks*, 1997.
- [14.] B. Das and V. Bharghavan, "Routing in Ad Hoc Networks Using Minimum Connected Dominating Sets", *International Conference on Communications*, 1997.
- [15.] R. Sivakumar, B. Das, and V. Bharghavan, "An Improved Spinebased Infrastructure for Routing in Ad Hoc Networks", *IEEE Symposium on Computers and Communications*, 1998.
- [16.] M. R. Garey, D. S. Johnson, "Computers and Intractability. A guide to the Theory of NP-completeness", *Freeman*, New York, 1979.
- [17.] K.M. Alzoubi, P.-J. Wan, and O. Frieder, "New Distributed Algorithm for Connected Dominating Set in Wireless Ad Hoc Networks", *Proceedings of the 35th Hawaii International Conference on System Sciences*, Hawaii, 2002.
- [18.] K.M. Alzoubi, P.-J. Wan, and O. Frieder, "Distributed Heuristics for Connected Dominating Sets in Wireless Ad Hoc Networks", *Journal of Communications and Networks*, vol. 4, no. 1, March 2002.
- [19.] H. Brey and D.G. Kirkpatrick. Unit disk graph recognition is NP-hard. *Computational Geometry. Theory and Applications*, 9(1-2):3–24, 1998.
- [20.] T.M. Chan. Polynomial-time approximation schemes for packing and piercing fat objects. *Journal of Algorithms*, 46(2):178–189, 2003.
- [21.] X. Cheng, X. Huang, D. Li, W. Wu, and D.-Z. Du. A polynomial-time approximation scheme for the minimum-connected dominating set in ad hoc wireless networks. *Networks*, 42:202–208, 2003.
- [22.] B. N. Clark, C. J. Colburn, and D. S. Johnson. Unit disks graphs. *Discrete Mathematics*, 86:165–177, 1990.
- [23.] I. Cidon and O. Mokryn, "Propagation and Leader Election in Multihop Broadcast Environment", *The 12th International Symposium on distributed Computing (DISC)*, 1998.
- [24.] I.J. Dejter and O. Serra, "Efficient dominating sets in Cayley graphs", *Discrete Applied Mathematics*, Vol. 129, (2003), pp. 319-328.
- [25.] T.W. Haynes, S.T. Hedetniemi and P.J. Slater, *Fundamentals of Domination in Graphs*, Marcel Dekker, (2000).
- [26.] Jia Huang and Jun-Ming Xu, "The bondage numbers and efficient dominations of vertex-transitive graphs", *Discrete Mathematics*, Vol. 308, (2008), pp. 571-582.
- [27.] S. Lakshminarayanan and S.K. Dhall, "Ring, torus and hypercube architectures/ algorithms for parallel

- computing", *Parallel Computing*, Vol.25,(1999), pp.1877-1906.
- [28.] J. Lee, "Independent perfect domination sets in Cayley graphs", *Journal of Graph Theory*, Vol.37, No.4, (2001), pp.213-219.
- [29.] N. Obradović, J. Peters and Goran Ružičić, "Efficient domination in circulant graphs with two chord lengths", *Information Processing Letters*, Vol.102, (2007), pp.253-258.
- [30.] T. Tamizh Chelvam and I. Rani, "Dominating sets in Cayley Graphs on  $Z_n$ ", *Tamkang Journal of Mathematics*, Vol.38, No.4, (2007), pp.341-345.
- [31.] T. Tamizh Chelvam and I. Rani, "Independent domination number of Cayley graphs on  $Z_n$ ", *The Journal of Combinatorial Mathematics and Combinatorial Computing*, Vol.69, (2009), pp.251-255.
- [32.] T. Tamizh Chelvam and I. Rani, "Total and Connected domination numbers for Cayley graphs on  $Z_n$ ", *Advanced Studies in Contemporary Mathematics*, Vol.20, (2010), pp.57-61.