Total dominating set based algorithm for connected dominating set in Ad hoc wireless networks

S. BALAJI
Department of Mathematics
SASTRA University
Thanjavur - 613 401
INDIA
balaji_maths@yahoo.com

K. KANNAN
Department of Mathematics
SASTRA University
Thanjavur - 613 401
INDIA
kkannan@maths.sastra.edu

Y. B. VENKATAKRISHNAN
Department of Mathematics
SASTRA University
Thanjavur - 613 401
INDIA
venkatakrish2@maths.sastra.edu

Abstract: In an efficient design of routing protocols in ad hoc wireless networks, the connected dominating set (CDS) is widely used as a virtual backbone. To construct the CDS with its size as minimum, many heuristic, meta-heuristic, greedy, approximation and distributed algorithmic approaches have been proposed in the recent years. These approaches mostly concentrated on deriving independent set and then constructing the CDS using Steiner tree and also these algorithms perform well only for the graphs having smaller number of nodes and also for the networks that are generated in an one fixed 2D simulation area. This paper provides a novel approach for constructing the CDS, based on the concept of total dominating set and bipartite theory of graphs. Since the total dominating set is the best lower bound for the CDS, the proposed approach reduces the computational complexity to construct the CDS through the number of iterations. Moreover the conducted simulation reveals that the proposed approach finds better solution than the recently developed approaches when all the three important factors of ad hoc network such as number of nodes, transmission radio range and area of network density varies.

Key–Words: connected dominating set, total dominating set, adhoc, algorithms.

1 Introduction

An ad hoc wireless network is a communication system without the aid of any fixed infrastructure. In an ad hoc wireless network collection of mobile hosts with wireless network may communicate each other from a temporary network, without the aid of any launched infrastructure or any particular administration. One can see application of ad hoc wireless networks in many diverse fields such as mobile commerce, search and rescue in a military battlefield. The problem concerning in the ad hoc wireless networks is the design of routing protocols for allowing communication between the hosts, but the nature of ad hoc networks makes this problem a challenging one. However virtual backbone of an ad hoc wireless network can be modeled as a computing connected dominating set in a graph where the network is considered as a graph, mobile hosts of the network treated as nodes of the graph. This model reduce the problem of ad hoc wireless network in to the well known minimum connected dominating set problem (MCDS) in graph theory.

For an undirected graph $G(V, E)$ with the vertex set V and the edge set E, a set S of vertices is a dominating set of G if every vertex not in S is adjacent to at least one member of S. If the subgraph of G induced by S is connected, then S is called connected dominating set (CDS). A connected dominating set of minimum cardinality is called minimum connected dominating set (MCDS) of the given graph G. While finding CDS(MCDS) in a general graph has shown to be NP-complete problem [4], it is worthy to note that finding CDS(MCDS) has many suitable applications. Many algorithms [2, 8, 9, 12, 14, 16, 19] have been proposed for finding a MCDS in ad hoc wireless network through virtual backbone.

The existing algorithms for constructing CDS can be divided into three sets. The first set of algorithms grows like a tree. In these algorithms the set S initially contains one node and some neighbours are added repeatedly into the set S until it is a CDS. The second set of algorithms divides the given network into different simple regions and correspondingly finds disconnected dominating tree for each regions and they join them finally using minimum spanning tree or Steiner tree. The third set of algorithms consider the whole node set as initial CDS and construct the final CDS by recursively removing the redundant nodes using Steiner tree, without affecting the dominators in the nodes.

The rest of the paper is as follows: A brief literature survey presented in Section II. In Section III all the necessary definitions and results related to the problem are stated. The pseudo-code, description and implementation of the proposed TD algorithm through an example problem are discussed in Section IV. In Section V simulation and results are given and section VI ends with our conclusion.

2 Related Works

Up-to-date references on the construction of connected dominating set, a virtual backbone, on ad hoc wireless networks until about 2009 are presented in [11]. Different routing protocol techniques for ad hoc wireless networks based on swarm intelligence, based on data-centric, hierarchical and location and based on their security aspects clearly surveyed in [1, 5, 17]. Other recent heuristic and meta-heuristic algorithms proposed for the construction of connected dominating set includes: Leu and Chang [9] developed a more recent version of weight value algorithm based on real time characteristic of mobile ad hoc networks, suitable for both static and dynamic environment. But this algorithm does not offer an improvement to the increases in the number of covered nodes and also in the radio range. Potluri and Singh [15] developed two meta heuristic algorithm namely hybrid genetic algorithm and hybrid ant colony optimization algorithm for the problem of computing minimum weight dominating set in unit disk graphs. Purohit and Sharma [16], based on the computation of convex hulls of sensor nodes, developed an algorithm for constructing MCDS in unit disk graphs. Yu et al [19] measured the quality of constructed CDS not only by the size of the CDS and also by another metric called CDS diameter. The metric diameter of a given connected graph is the length of the longest shortest paths between a pair of nodes in the graph. Based on this metric they developed a heuristic for constructing connected dominating sets with minimum size. In [14] Misra and Mandal proposed a new heuristic called collaborative cover using the graph theoretic concept, domatic number of a connected graph. Moreover the following principle used to define the optimal substructure i.e. subset of independent dominator preferably with a common connector considered as optimal substructure. They claimed that their heuristic is better than that of degree based heuristics of CDS. Lu et. al. [12] developed an approximation algorithm for the construction of CDS in UDG using some geometric features like angle and area of nodes distribution in UDG. Han [6] combined the concept of zone and level to scatter the CDS, virtual backbone of a wireless network. Based on these ideas they partition the wireless network into different zones. For each zone a dominating tree constructed and they are connected by inserting additional connectors at the zone borders, it produces the final CDS.

Construction of CDS had also been focused by various greedy heuristics such as a polynomial time greedy heuristic developed in [13] focused on most stable routes in order to get minimum number of route transitions. This algorithm make use of the complete knowledge about location of nodes that change in near future and also concentrated on the change of the communication structure (path, trees, connected dominating set etc.,) should be in minimum number. A greedy algorithm based on the concept of steiner tree proposed in [10] which construct a CDS within a factor of $4.8 + log5$, a better performance ratio than previously developed and some neighbourhood based local search approach developed in [3].

Other than that of heuristic, meta heuristics and greedy approaches, commonly used algorithms in the construction of CDS are distributed algorithms. Here are some of the recently developed distributed algorithms. Islam et al. [8] presented a distributed algorithm that computes a family of non-trivial connected dominating sets (CDS) with the goal to minimizing the number of frequencies of each node in these sets. Alzoubi et al. [2] developed their own distributed algorithm with two phases which construct a maximal independent set and a dominating tree respectively. They have claimed that their algorithm has an approximation factor of at most 8, $O(n)$ time complexity and $O(nlogn)$ message complexity. In recent Yin et al. [18] proposed a single phase distributed algorithm called DSP-CDS for constructing a connected dominating set. In this algorithm each node get the information from its one-hop neighbourhood, using this it makes a local decision that whether to join the dominating set or not. They also guaranteed that at the end, the algorithm produces connected dominating set because of each node bases its decision on a key variable.

Major drawback of all the available techniques includes: most of them initially find independent set using different approaches and then finds CDS using Steiner tree and they never use the properties of dominating set, the construction of MCDS using some more complicated strategies such as geometric properties of distribution of nodes in the independent set and some more complicated logical relations between the nodes of a graph. These types of method of algorithms causes more time delay and more energy consumption in the construction of MCDS, because of this they fail to maintain their actual performance guarantee. Moreover in the reported computational experiments of MCDS of existing algorithms, they were tested their algorithms for small number of mobile hosts and also they particularly considered only one square area to generate different random ad hoc networks. Due to various factors around the world, change of location (area) of network is unavoidable one. Similarly we cannot restrict the number of mobile hosts into smaller number. Therefore these factors are very important to judge the performance of one algorithm, developed for constructing the CDS.

By considering the above conditions and to make simple and efficient heuristic, in this paper, we have proposed our own new heuristic algorithm based on the parameter total dominating set of a graph. Since the total dominating set of a graph is the very best lower bound to CDS of the graph, finding CDS through total dominating set gives very low computational complexity than other approaches where they constructed CDS through dominating set or independent set. The proposed approach also has the other advantages like easy to implement, reduced computational complexity, less set up cost and provides good solutions when compared to other existing algorithms in the literature.

3 Preliminaries

Let $G = (V, E)$ be an undirected simple graph, where $V = \{v_1, v_2, ..., v_n\}$ is the set of vertices and $E \subseteq V \times V$ (not in ordered pairs) is the set of edges with cardinality of $|V| = n$ and $|E| = m$ and the complement graph of $G(V, E)$ is the graph $\overline{G}(V, \overline{E})$, where $\overline{E} = \{(v_i, v_j) \in V, v_i \neq v_j \text{ and } (v_i, v_j) \notin E\}$. Then we have the following basic definitions relative to the forthcoming chapters:

Connected graph: A graph G is said to be connected if there is a path between every pair of distinct vertices of a graph G. A graph which is not connected is called disconnected graph.

Neighborhood of a vertex: For each $v \in V$, the neighborhood of v is defined by $N(v) = \{u \in V/u \text{ is adjacent to } v\}$ and the closed neighbourhood of v is defined by $N[v] = \{v\} \cup N(v)$.

Degree of a vertex: The degree of a vertex $v \in V$, denoted by $d(v)$ and is defined by the number of neighbors of v i.e., $d(v) = |N(v)|$.

Dominating set: A dominating set for a graph $G(V, E)$ is a subset D of V such that every vertex not in D is adjacent with atleast one member of D. The minimum cardinality of a dominating set is denoted by $\gamma(G)$ and is called the domination number of G.

Minimal dominating set: Minimal dominating set (mDS) is a dominating set (DS) such that any proper subset of mDS is not a DS; in other words, for any $v \in$ mDS either v is an isolate of the mDS or there exists at least one node $u \in V$- mDS such that u is not dominated by any node in mDS except v.

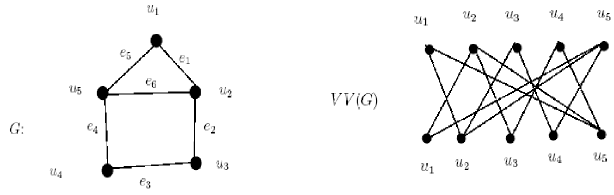

Figure 1: Graph G and its corresponding bipartite graph $VV(G)$

Connected dominating set: A dominating set D of a graph $G(V, E)$ is said to be a connected dominating set (CDS) if the subgraph induced by D is connected.

Minimal connected dominating set: Minimal connected dominating set $(mCDS)$ is a CDS such that removing any node from this set will make it no longer a CDS.

k-Connected k-Dominating set: A vertex set $D \subseteq V$ is a k-dominating set(or simply $k - DS$) of G if every vertex not in D has at least k neighbouring vertices in D.

A $k - DS$ is a k-connected k-dominating set (or simply $k - CDS$) of G if the subgraph $G[D]$ induced from D is k-vertex connected.

Total dominating set: A subset $S \subseteq V$ is a total dominating set if for every $u \in V$ there exists $v \in S$ such that u and v are adjacent. A subset $S \subseteq V$ is called a minimal total dominating set if no proper subset of S is a total dominating set. The minimum cardinality of a minimal total dominating set is called the total domination number of G and is denoted by $\gamma_t(G)$.

Bipartite theory of graphs was introduced by Stephen Hedetniemi and Renu Laskar [7]. It has been mentioned that many of the concepts in graph theory has equivalent formulations as concepts for bipartite graphs. One such formulation is the Y-dominating set of a bipartite graph.

Y-dominating set: Let $G' = (X, Y, E)$ be a bipartite graph. A subset D of X is a Y-dominating set if for every $y \in Y$, there exists $x \in D$ such that x and y are adjacent. The Y-domination number of G' denoted by $\gamma_Y(G')$ is the minimum cardinality of a Y-dominating set.

From a graph $G = (V, E)$ one can construct bipartite graph $VE(G)$, $EV(G)$, $VV(G)$ and $VV^+(G)$. Here we give the method to construct $VV(G)$ as given in [7].

Bipartite graph $VV(G)$***:*** Let V' be a copy of V. The bipartite graph $VV(G) = (V, V', E')$ where $E' = \{uv' : uv \in E\}$.

Example 1 *The graph G and its corresponding bipartite graph $VV(G)$ is given Fig. 1.*

Theorem 2 *For any graph G, $\gamma_Y(VV(G)) = \gamma(G)$ [7].*

Remark 3 *In a connected graph $G = (V, E)$, every total dominating set is a dominating set of G. Also every connected dominating set is a total dominating set. Hence, $\gamma(G) \leq \gamma_t(G) \leq \gamma_c(G)$.*

From the above remark, total domination number of a graph is a better lower bound to MCDS. Hence, finding the total dominating set reduces the time complexity in terms of number of iterations.

4 Proposed TD Algorithm

The following algorithm is designed to find the CDS of a graph. The proposed algorithm is divided into three phases. In the first phase total dominating set is constructed and to make the total dominating set as connected one, in the second phase connector nodes are found with the help of neighborhood based selection criteria. In the final phase exhaustive local search procedure in applied to reduce the number of nodes in the CDS, make it as an optimal minimum connected dominating set.

Algorithm 1: *Proposed Total dominating (TD) set based algorithm for CDS*

Input: A connected graph $G(V, E)$ with $|V| = n$ and $|E| = m$.

Output: Minimum connected dominating set $D \subseteq V$.

Intialization:

$D \leftarrow \phi$; $D^{'} \leftarrow \phi$;

begin

- For the graph $G(V, E)$, corresponding bipartite graph $VV(G)$ is constructed.

- $VV(G) = (X, Y, E')$ where $X = V, Y = V'$ is a copy of V and $E' = \{(x, y')/(x, y) \in E\}$

- $D' \leftarrow Y - dom(X, Y, E')$

- $D \leftarrow D'$

- Partition the set D' into subsets $D_1, D_2, ..., D_k$ such that $\exists$ a path between any two vertices in $D_i, i = 1, 2, ..., k$

repeat

for $i \leftarrow\ 1\ to\ k$

- $B_i = \{v \in V - D/N(v) \cap D_i \neq \phi\}$

end

- $C_\alpha \leftarrow$ any one of collection of $\binom{k}{\alpha}$ $D_i, i = 1, 2, ..., k$

- Search for the common element $v \in \bigcap_{p \in C_\alpha} B_p$ based on the sets of combination $p \in C_\alpha$

- $D \leftarrow D \cup \{v\}$

- combine all the set elements in C_α together with the last set element of the combination and also add the vertex into the set. i.e., suppose if $v \in B_1 \cap B_2$ for one combination of the sets $D_1 D_2$ then $D_2 = D_2 \cup D_1 \cup \{v\}$

- $k \leftarrow k - \alpha + 1$

until $\bigcap_{p \in C_\alpha} B_p \neq \phi$

while D is not connected ***do***

for $i \leftarrow\ 1\ to\ k - 1$

for $j \leftarrow\ i + 1\ to\ k$

- search for $N(D_i) \bigcap N(D_j) = \phi$

- then A

$= \{v \in V - D / \left(N^2(D_i \bigcap N(v))\right) \bigcap \left(N^2(D_j \bigcap N(v))\right)\}$

- $\forall w \in N(v) \bigcap A$ for every $v \in A$

- search for

- $D \leftarrow D \cup \{v, w\}$

end

end

end

- *local search*$(V, E, G[D])$

return MCDS $\leftarrow D$

end

Algorithm 2: $Y - dom(X, Y, E')$

Input: bipartite graph $VV(G)$ of given graph $G(V, E)$

Output: y - dominating set $VV(G)$ = Total dominating set of $G(V, E)$

do

- $v \leftarrow \max_{v \in X} N(v)$

- $D \leftarrow D \cup \{v\}$

- $X \leftarrow X - \{v\}$

- $Y \leftarrow Y - N(v)$

while $Y \neq \phi$

Algorithm 3: *local search*$(V, E, G[D])$

while D is connected ***do*** $\forall v \in D$

- $B' = \{N[w]/w \in D - \{v\}\}$

if $N[v] - B' = \phi$

- then $D \leftarrow D - \{v\}$

else

- $D \leftarrow D$

end

end

The algorithm operates as follows: The proposed algorithm proceeds in three phases. Initially the minimum connected dominating set D is empty and total dominating set D' of a graph G is empty.

In the first phase of the algorithm, for a given graph G the corresponding bipartite graph $VV(G)$ is constructed and its Y-dominating set is found. Procedure of finding Y-dominating set of a graph is described in Algorithm 2. By the Theorem 3.1, we get the total dominating set D' of a graph G. Then D is initialized as D'.

In the second phase, the TD algorithm partitions the set D' in to subsets $D_1, D_2, ..., D_k$ where each D_i's are connected. Then we search for a common element $v \in V - D$ such that $N(v) \cap D_i \neq \phi, 1 \leq i \leq k$. If such an element exists, then D is updated as $D \cup \{v\}$. Otherwise, we repeat the above procedure for $\binom{k}{\alpha}$ combinations of D_i's where $2 \leq \alpha \leq k-1$. For any one of the combination in $\binom{k}{\alpha}$ say p, the above condition is satisfied then the corresponding element $v \in V - D$ is added into the set D. From the above if $D_1, D_2, ..., D_p$ along with v is connected, we assign D_p with $D_p \cup \{v\}$. Now, $D_p, D_{p+1}, ..., D_k$ forms a partition of the set D'. This process is repeated until there is no common element connecting at least two of the subsets. Then the algorithm search for the disconnected sets D_i and D_j using the criteria $N(D_i) \bigcap N(D_j) = \phi$ where $i = 1$ to $k-1$ and $j = i+1$ to k. Then by neighborhood search procedure technique, the TD algorithm finds two adjacent vertices in $V - D$ such that one vertex is adjacent to D_i and the other vertex is adjacent to D_j . By this way, all D_i's are connected.

In the third phase, we drop the redundant elements in the set D to get a MCDS set using the exhaustive local search procedure which is described in Algorithm 3.

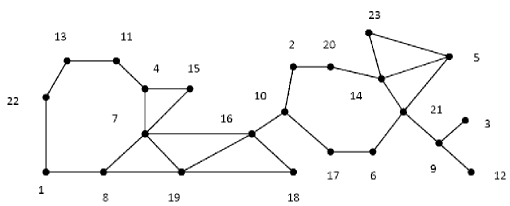

Figure 2: Initial topology of the network

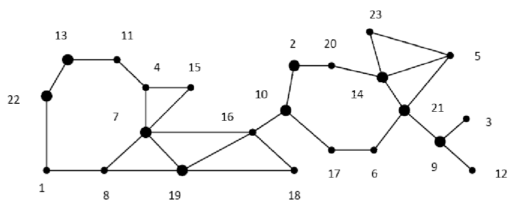

Figure 3: Total dominating set of the network

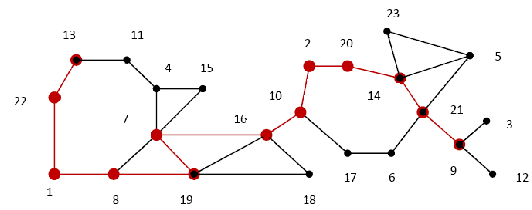

Figure 4: More connectors are selected

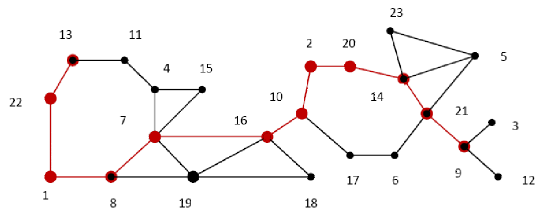

Figure 5: MCDS of the network

Example 4 *The problem in [6] is considered as an example problem to show the procedure of the proposed algorithm. This example shown in the Fig. 2, 3, 4 and 5. Fig 2 shows the topology of network taken. In which the set of vertices in the graph below with larger radius represents the total dominating set. The total dominating set is partitioned into four subsets* $D_1 = \{7, 19\}, D_2 = \{13, 22\}, D_3 = \{9, 14, 21\}$ *and* $D_4 = \{2, 10\}$, *which is shown in the Fig 3. Based on the TD algorithm procedure, now the vertex 16 is added to the total dominating set. Now,* $D_2 = \{13, 22\}, D_3 = \{9, 14, 21\}$ *and* $D_4 = \{2, 7, 10, 16, 19\}$. *Similarly the vertex 20 is added to the total dominating set, which is shown in the Fig 4. Further the vertex 19 is removed using the exhaustive local search procedure to get a MCDS set and final MCDS set shown in the Fig 5.*

Theorem 5 *The proposed TD algorithm returns a connected dominating set, and has a time complexity of* $O(nm)$.

Proof. Let $G(V, E)$ be an undirected connected graph. By the algorithm procedure, it is clear that first phase of the algorithm produces a total dominating set D' and in the second phase of the algorithm the set D' partitioned into smaller subsets such that each subset is a connected set. Moreover every subset is connected to all the other sets through some other sets or directly by adding connector nodes between them using the neighborhood search procedure in the second phase. The third phase of the algorithm removes some more vertices from D and produces updated CDS D. Therefore it is enough to prove that updated D is still a CDS. To take out or maintain a vertex from D in the third phase, at each iteration, for every $v \in D$ if the difference between the sets $N[v]$ and $\{N[w]/w \in D - \{v\}\}$ is empty, we can find at least one adjacent vertex in $D \leftarrow D - \{v\}$ neighbor to all the closed neighboring elements of v. This implies that updated D is still a connected dominating set. If these selected vertices fail to satisfy the condition described in the Algorithm 3 i.e. phase three of the TD algorithm it implies that if we remove any one of the vertex from the set, obtained from the second phase, it becomes a disconnected one. Therefore in this case the proposed algorithm returns the set of the second phase as a final CDS; in this case also D is a CDS.

The computational complexity of the algorithm is as follows: In Algorithm 1 while loop is executed at most n times. Adding or removing the vertices at each step of the Algorithm 2, determines the connectivity of a graph, to do this, it will be executed at most $O(m(n-d))$ times where d represents the size of the dominating set. To remove redundant nodes in the CDS, obtained by Algorithm 1 and Algorithm 2, Algorithm 3 will perform its procedures at most d times. Thus, the computational complexity of the TD algorithm is given by $O(mn)$. □

5 Simulation Results

To test the performance of the proposed algorithm constructing the CDS for some random networks, we have conducted extensive simulations. These simulations divided into three sets based on the main parameters of a network such as number of nodes, transmission radio range and area density of the network and TABLE I gives this information in brief.

Table 1: Parameters position in all the three simulations

$Sim.$	n	r	Area
1.	Varies	150m, 200m	1000m X1000m
2.	300, 600	Varies	1000m X 1000m
3.	50, 100	200m	Varies

This section shows the results of conducted simulations. Simulations were carried out on a Intel Pentium Core2 Duo Processor PC having 1.6GHz CPU and 1GB RAM. All the procedures of the TD algorithm have been coded and implemented in MATLAB. To carry out the effectiveness of the proposed TD algorithm, comparison made with the three recently developed algorithms of MCDS presented in [6, 14, 19] and for our convenience those algorithms are noted such as WVA [6], JY [19] and CH [14]. In all the simulations carried out in this paper, the following condition is constantly implemented to generate random network instances i.e. number of nodes considered is uniformly distributed in a 2D simulation area of size length × length in some unit measurement at random and the link between two nodes are established if the distance between the nodes are not longer than r (transmission range) units.

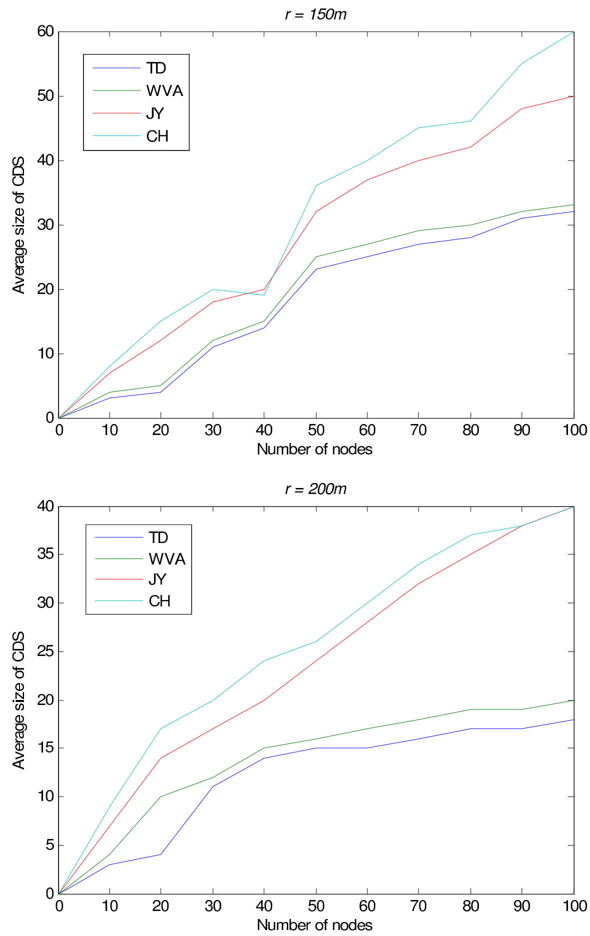

Figure 6: Comparison of average size of CDS for different number of nodes

To evaluate the performance of the proposed algorithm under various number of nodes, the corresponding n number of nodes randomly deployed in 1000m × 1000m 2D simulation area. n varied from 10 to 100 in the interval increment of 10. Each node has been assigned to a fixed transmission radio range 150m. For each fixed number of nodes and the transmission radio range, 1000 network instances were created. Before start of the simulation, all the networks are checked to make sure of that their connectivity. All the four algorithms were ran on the 1000 network instances of each node and the corresponding CDS size was noted. The above procedure is repeated by assigning another fixed transmission radio range 200m. Average size of CDS is taken as the size of the CDS produced by each algorithm and the obtained results are shown in the Fig. 6. From the results given in Fig. 6, it is clear that, for all the four algorithms the size of the CDS increases when the number of nodes in the network increases. Moreover from the obtained results we observed that the proposed TD algorithm find better average results than other compared algorithms.

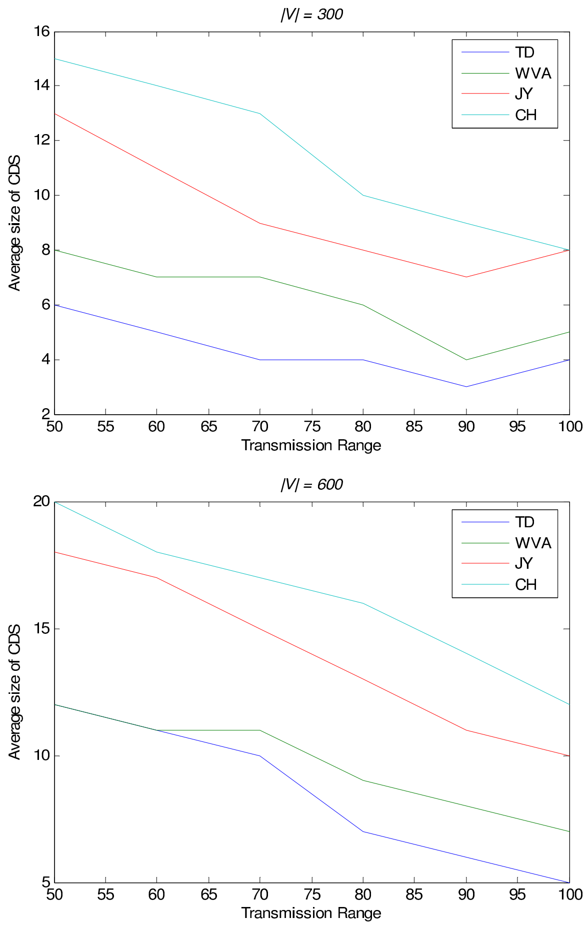

Figure 7: Comparison of average size of CDS for different transmission ranges

In the second set of simulation, we compare the size of CDS of all the four algorithms when the transmission radio range varies. Through this simulation we can judge the performance of proposed algorithm over different transmission ranges and how it will affect the size of the CDS. In this simulation initially 300 nodes are randomly distributed into a fixed area of size 1000m $\times$ 1000m. Each node has been assigned to a transmission range starting from 50m, with the interval difference of 5m, each node further has been assigned transmission ranges up to 100m. For each n and r, 1000 network instances were created and simulations are carried out on all these instances. The same process is repeated for another set of 600 nodes, randomly deployed in the same area. The average size of CDS constructed by each algorithm for two different set of nodes of different transmission ranges shown in the Fig. 7. As expected, from the Fig. 7, we can see that the algorithms JY [19] and CH [14] deviated highly form the constructed size of the CDS when compared to WVA [6] and the proposed TD algorithm.

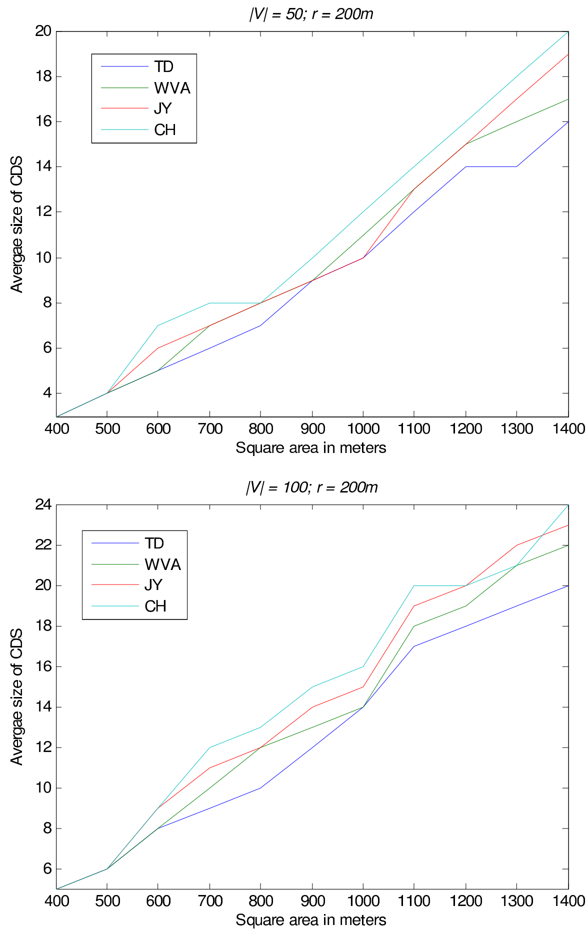

Figure 8: Comparison of average size of CDS in different area densities

Simulations were also carried out to compare the effectiveness of the proposed algorithm when the area of the network density is varied. It is important to note that there are no similar simulations carried out in the previous literature so far. In this simulation for two different sets of nodes $n = 50$ and $n = 100$, the transmission range of each node was fixed as $r = 200$m. The two sets of nodes are randomly deployed in the different 2D simulation area from 200m $\times$ 200m to 1000m $\times$ 1000m with the interval increment of 100m. 1000 network instances were generated for each described values of n, r and area and all the four algorithms ran on these instances to construct the CDS and the average size of CDS obtained by each algorithm shown in the Fig. 8. From the obtained results we observed that the proposed TD algorithm outperformed recently developed heuristics not only in the different number of nodes and different transmission ranges but also in the different area network density

6 Conclusions

In this paper a novel approach called TD algorithm is established to construct the CDS, a virtual backbone of ad hoc wireless networks, based on the total dominating set and bipartite theory of graphs. In the previous research approaches of construction of connected dominating set, most of them based on the independent set construction, some complicated strategies were applied in few papers and some follows other techniques which are not even contains any single graph theoretic ideas even though CDS is one of the well known graph optimization problem. In this paper the proposed TD algorithm purely based on the relation between the total dominating set, Y-dominating set of bipartite graph and the CDS. The total dominating set is best lower bound for the CDS than the dominating set, the proposed approach give the guarantee to reduce the computational complexity. The conducted simulation over the different important factors such as smaller to larger nodes, transmission ranges and area of network density reveals that the proposed approach outperforms the recently developed approaches in the construction of CDS. The above mentioned advantages and the simplicity of the proposed heuristics make them an attractive alternative approach for solving the graph optimization problems in dynamic environments. As a future work, we plan to conduct extensive simulation study on the performance of MCDS in carrying out important tasks such as routing and area monitoring.

References:

[1] K. Akkaya, M. Younis, A survey on routing protocols for wireless sensor networks, *Ad Hoc Networks*, 3 (2005), pp.325-349.

[2] K. M. Alzoubi, P-J. Wan, O. Frieder, Distributed Construction of Connected Dominating Set in Wireless Ad Hoc Networks, *Mobile Networks and Applications*, 9 (2004), pp.141-149.

[3] S. Balaji, N. Revathi, An efficient heuristic for the minimum connected dominating set problem on Ad hoc wireless networks, *Proceedings of WASET*, 68 (2012), pp.1045-1051.

[4] M. R. Garey, D. S. Johnson, *Computers and Intractability. A Guide to the theory of NP-Completeness*, W. H. Freeman. 1979.

[5] A. K. Gupta, H. Sadawarti, A. K. Verma: A review of routing protocols for mobile addoc networks, *WSEAS Transaction on communication*, 10 (2011), pp.331-340.

[6] B. Han, Zone-based virtual backbone formation in wireless ad hoc networks, Ad hoc Networks, 7 (2009) 183-200.

[7] S. T. Hedetniemi, R. Laskar, A Bipartite theory of graphs II, *Congressus Numerantium*, 64 (1988), pp.137-146.

[8] K. Islam, S.G. Akl, H. Meijer, Distributed Generation of a Family of Connected Dominating Sets in *Wireless Sensor Networks*, B. Krishnamachari et al. (Eds.): DCOSS 2009, LNCS 5516, (2009), pp.343-355, Springer-Verlag Berlin Heidelberg.

[9] S. Leu, R-S. Chang, A weight-value algorithm for finding connected dominating sets in a MANET, *Journal of Network and Computer Applications*, 35 (2012), pp.1615-1619.

[10] Y. Li, M.T. Thai, F. Wang, C-W. Yi, P-J. Wan, D-Z. Du, On greedy construction of connected dominating sets in wireless networks, *Wirel. Commun. Mob. Comput.*, 5 (2005), pp.927-932.

[11] Z. Liu, B. Wang, L. Guo, A survey on connected dominating set algorithm for wireless sensor networks, *Information Technology Journal*, 9 (6) (2010), pp.1081-1092.

[12] G. Lu, M.-T. Zhou, Y. Tang, M-Y. Zhao, X.-Z. Niu, K. She, Approximation algorithms for connected dominating set problem in unit disk graphs, *Journal of electronic science and technology of China*, 7(3) (2009), pp.214-222.

[13] N. Meghanathan, A. Farago, On the stability of paths, Steiner trees and connected dominating sets in mobile ad hoc networks, *Ad Hoc Networks*, 6 (2008), pp.744-769.

[14] R. Misra, C. Mandal, Minimum connected dominating set using a collaborative cover heuristic for Ad Hoc Sensor Networks, *IEEE Transactions on Parallel and distributed systems*, 21(3) (2010), pp.292-302.

[15] A. Potluri, A. Singh, Hybrid metaheuristic algorithms for minimum weight dominating set, *Applied Soft Computing*, Article in press.

[16] G. N. Purohit, Usha Sharma, Constructing Minimum Connected Dominating Set: Algorithmic approach, *International journal on applications of graph theory in wireless ad hoc networks and sensor networks* (GRAPH-HOC), 2(3) (2010), pp.59-66.

[17] M. Saleem, G. A. D. Caro, M. Farooq, Swarm intelligence based routing protocol for wireless sensor networks: Survey and future directions, *Information Sciences*, 181 (2011), pp.4597-4624.

[18] B. Yin, H. Shi, Y. Shanga, An efficient algorithm for constructing a connected dominating set in mobile ad hoc networks, *J. Parallel Distrib. Comput.*, 71 (2011), pp.27-39.

[19] J. Yu, N. Wang, G. Wang, Heuristic Algorithms for Constructing Connected Dominating Sets with Minimum Size and Bounded Diameter in *Wireless Network*, G. Pandurangan et al. (Eds.): WASA 2010, LNCS 6221, (2010), pp.11-20, Springer-Verlag Berlin Heidelberg.