Approximation Algorithmic Performance for CEDS in Wireless Network

Venugeetha Y (Asso. Prof.),
Research Scholar
Department of Computer Science and Engineering,
Global Academy of Technology,
Bengaluru -560098,
Karnataka, India.
venugeeta@gmail.com

Dr. B P Mallikarjunaswamy,
(Professor), Department of Computer Science and Engineering,
Sri Siddhartha Institute of Technology, Maraluru, Tumkuru - 572105, Karnataka, India.
drbpmswamy@rediffmail.com

Dr. C D Guruprakash
Professor
Dept. of Computer Science and Engineering
Sri Siddhartha Institute of Technology, Tumkur-572102
mail id guruprakashcd@ssit.edu.in

Abstract— A well-organized design of routing protocols in 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 anticipated. These approaches are concentrated on deriving independent set and then constructing the CDS using UDG, Steiner tree and these algorithms perform well only for the graphs having smaller number of nodes. For the networks that are generated in a fixed simulation area. This paper provides a novel approach for constructing the CDS, based on the concept of total edge dominating set. 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. The conducted simulation reveals that the proposed approach finds better solution than the recently developed approaches when important factors of network such as transmission radio range and area of network density varies.

Keywords— *connected dominating set, total edge dominating set.*

*****

I. Introduction

A wireless network is a communication system without any fixed infrastructure. In a wireless network collection of wireless 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 wireless networks in many diverse fields such as search and rescue in a military battlefield. The problem concerning in the wireless networks is the design of routing protocols for allowing communication between the hosts, but the nature makes it a challenging one. However virtual backbone of a wireless network can be modeled as a computing connected dominating set in a graph where the network is considered as a graph, hosts of the network treated as nodes of the graph. The result reduces the problem of wireless network in to the well known minimum connected dominating set problem (MCEDS) 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).

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

II. Literature Review

Guha and Khuller first proposed two polynomial time algorithms to construct a CDS in a general graph G. These algorithms are greedy and centralized. The first one has an approximation ratio of 2(H(Δ)+1), where H is a harmonic function. The idea of this algorithm is to build a spanning tree rooted at the node that has a maximum degree and grow a tree until all nodes are added to the tree. The non-leaf nodes in the tree form a CDS. In particular, all the nodes in a given network are white initially. The greedy function that the algorithm used to add the nodes into the tree is the number of the white neighbors of each node or a pair of nodes. The one with the largest such number is marked black and its neighbors are marked grey. These black and grey nodes are then added into the tree. The algorithm stops when no white node exists in G. The second algorithm is an improvement of the first one. This algorithm consists of two phases. The first phase is to construct a dominating set and the second phase is to connect the dominating set using an approximation algorithm for Steiner tree problem. With such improvement, the second algorithm has a performance ratio of H(Δ)+2 [1].

Mathieu Couture et. al, Given a graph G, a k-dominating set of G is a subset S of its vertices with the property that every vertex of G is either in S or has at least k neighbors in S. We present a new incremental local algorithm to construct a k-dominating set. The algorithm constructs a monotone family of dominating sets $D1 \subseteq D2 \ldots \subseteq Di \ldots \subseteq Dk$ such that each Di is an i-dominating set. For unit disk graphs, the size of each of the resulting i-dominating sets is at most six times the optimal [2].

Faisal N. Abu-Khzam et. al., considering the problem of dominating set-based virtual backbone used for routing in asymmetric wireless ad-hoc networks. These networks have non-uniform transmission ranges and are modeled using the well-established disk graphs. The corresponding graph theoretic problem seeks a strongly connected dominating-absorbent set of minimum cardinality in a digraph. A sub-set of nodes in a digraph is a strongly connected dominating-absorbent set if the subgraph induced by these nodes is strongly connected and each node in the graph is either in the set or has both an in-neighbor and an out-neighbor in it.

Distributed algorithms for this problem are of practical significance duet o the dynamic nature of ad-hoc networks. A first distributed approximation algorithm, with a constant approximation factor and O(Diam) running time, where Diam is the diameter of the graph. Moreover a simple heuristic algorithm and conduct an extensive simulation study showing that our heuristic outperforms previously known approaches for the problem [3].

Topology control is a fundamental issue in wireless ad hoc and sensor networks. Due to intrinsic characteristic of flatness, hierarchical topology can achieve the scalability and efficiency of a wireless network. To solve this problem, one can construct a virtual backbone network by using a connected dominating (CDS) set of a wireless network. In past few years, efficiently and fast construct a CDS in a wireless network as a virtual backbone has been the main research problem in hierarchical topology control [4].

Decheng Dai et al., Study the minimum weight dominating set problem in weighted unit disk graph, and give a polynomial time algorithm with approximation ratio 5+ε, improving the previous best result of 6+ε, A better constant-factor approximation for weighted dominating set in unit disk graph. Combining the common technique can compute a minimum weight connected dominating set with approximation ratio 9+ε, beating the previous best result of 10+ε in the same work [5].

Ning Chen et. Al., this paper, studies the Dominating Set problem with measure functions, which is extended from the general Dominating Set problem, corresponding problems on complexity, approximation and inapproximability for Dominating Set problem with measure functions [6].

Samir Khullerz, Sudipto Guhay, The dominating set problem in graphs asks for a minimum size subset of vertices with the following property: each vertex is required to either be in the dominating set or adjacent to some node in the dominating set; focus on the question of finding a connected dominating set of minimum size, where the graph induced by vertices in the dominating set is required to be connected as well. The problem arises in network testing, as well as in wireless communication.

Two polynomial time algorithms that achieve approximation factors of O(H(Δ)) are presented, where Δ is the maximum degree, and H is the harmonic function. This question arises in relation to the traveling tourist problem, where one is looking for the shortest tour such that each vertex is either visited, or has at least one of its neighbors visited. A generalization of the problem when the vertices have weights, and give an algorithm which achieves a performance ratio of 3 *ln* n. Also consider more general problem of finding a connected dominating set of a specified subset of vertices and provide an O(H(Δ)) approximation factor. To prove the bound, develop an optimal approximation algorithm for the unit node weighted Steiner tree problem [7].

Khaled M et. al., connected dominating set (CDS) has been proposed as virtual backbone or spine of wireless ad hoc networks. Three distributed approximation algorithms have been proposed in the literature for minimum CDS. To reinvestigate their performances algorithms have no constant approximation factors. Thus these algorithms cannot guarantee to generate a CDS of small size. Their message complexities can be as high as $O(n^2)$, and their time complexities may also be as large as $O(n^2)$and $O(n^3)$. A distributed algorithm that outperforms the existing algorithms, has an approximation factor of at most 8, O(n) time complexity and O(n log n) message complexity. By establishing the Ω (n log n) lower bound on the message complexity of any distributed algorithm for nontrivial CDS, algorithm is thus message-optimal [8].

A. Dominating Tree construction.

This phase constructs a tree spanning all the black nodes, referred to as *dominating tree*. All nodes in this dominating tree form a CDS. The dominating tree is initially empty. The root joins the dominating tree first. When each black node joins the dominating tree, it sends an invitation to all black nodes that are two hops away and outside the current dominating tree to join the dominating tree. This invitation will be relayed through the gray nodes. Each black node will join the tree when it receives the invitation for the first time together with the gray node which relays the invitation to itself. This process should be repeated until all black nodes are in the tree. The next is the implementation detail:

- The root sends an INVITE message.
- When a gray node receives for the *first* time an INVITE message from a black neighbor, it stores the ID of this black neighbor in its local variable inviter, and then relays such INVITE message.

International Journal on Future Revolution in Computer Science & Communication Engineering
Volume: 4 Issue: 3

ISSN: 2454-4248
213 – 219

- When a black node receives for the *first* time an INVITE message from a gray neighbor, it puts this gray neighbor as its parent in the dominating tree, then sends back a JOIN message towards this gray neighbor and finally initiated an INVITE message.
- Whenever a gray node receives a JOIN message towards itself from a black neighbor, it puts this black neighbor as its child. In addition, upon the receiving of the *first* JOIN message towards itself, it sends a JOIN message towards the black neighbor whose ID is stored in the local variable inviter.
- Whenever a black node receives a JOIN message towards itself from a gray neighbor, it puts this gray neighbor as its child.

Theorem 7 guarantees that whenever there is any black node outside the current dominating tree, at least one black node would join the dominating tree. Thus eventually all black nodes will join the dominating tree. A reporting process described as follows, if necessary, can be performed along the spanning tree T to notify the root of the completion. A gray node reports a COMPLETE message to its parent in the spanning tree if all has received a COMPLETE message from each child in the spanning tree. A black node reports a COMPLETE message to its parent in the spanning tree if all has received a COMPLETE message from each child in the spanning tree and itself has joined the dominating tree.
Note that each black node initiates one INVITE message and
one JOIN message (except the root); each gray node relays one INVITE message and at most one JOIN message. So the construction of the dominating tree requires $O(n)$ messages and(n) time. The same is true for the optional reporting process. Therefore, the total message complexity and time complexity of our algorithm are O (n log n) and O (n) respectively.
Finally, bound with the size of the dominating tree. Since each gray node appearing in the dominating tree is the parent of at least one black node, the total number of gray nodes in the dominating tree is at most one less than the number of black nodes. From Lemma 8, the total number of nodes in the dominating tree is at most.

$$2\,(4opt + 1)\,\Delta 1 = 8opt + 1:$$

In summary, we have the following performance results of the
distributed algorithm in [12].

III. Auxiliary Definition

Let G = (V,E) be an undirected simple graph, where V = {v1, v2, ..., vn} is the set of vertices and E ⊆ V × 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 G(V,E), where E = {(vi, vj) ∈ V, vi ≠ vj and (vi, vj) ∉ E}.

Below given are few terminology relevant to the paper:

2*(i)* Dominating Set –A graph G = (V,E) is a subset D of the Vertex Set V such that each vertex *v* is either in D or adjacent to some vertex *v* in D. This set contains Dominators which are the elements of dominating set. Examples of dominating set in a graph G are as shown below:

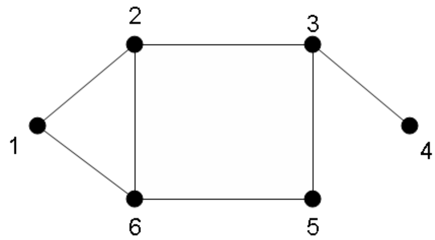

Fig 1: Dominating Sets are {1, 3}, {2, 3, 5} and {1, 2, 3, 4}.

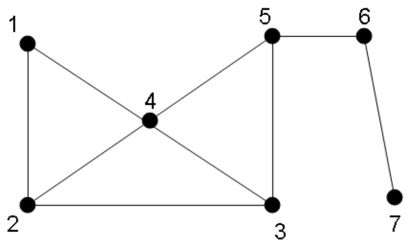

Fig 2: Dominating Sets are {1, 5, 7}, {4, 6} and {4, 5, 6}.

2*(ii)* Connected Dominating Set (CDS) – CDS of a graph G has a set of vertices with two properties:

1. D is a dominating set in G.
2. D induces a connected subgraph of G.

In Fig.1, {2, 3, 5} and {1, 2, 3, 4} are C D S. Similarly in Fig. 2, {4, 5, 6} is a C D S.

2*(iii)* Minimum Connected Dominating Set MCDS – MCDS is a connected dominating set with smallest possible cardinality among all the CDS of G. As in Figs. 1 and {2, 3, 5} and {4, 5, 6} are Minimum Connected Dominating Sets respectively.

2*(iv)* Independent Set – Graph G is a subset of the set of vertices such that no two vertices are adjacent in the subset. For example in Fig.1 {1, 3}, {1, 4, 5}, {2, 4, 5} are independent sets. In Fig.2 {1, 7}, {1, 4, 6, 7}

2*(v)* Maximal Independent Set – A set, which is not a subset of any other independent set i.e. it is a set S such that every edge of the graph has at least one end point not in S and every vertex not in S has at least one neighbor in S.

Six different Maximal Independent Set of following cubic graph are {1, 5, 7}, {3, 8}, {2,5}, {4,7}, {1,6} and {4, 6, 8, 2}.

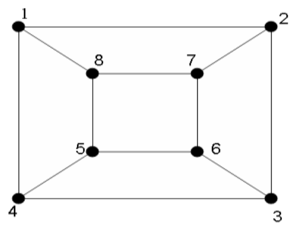

Figure 3: MIS in Cubic Graph

2*(vi)* Unit Disk Graph – A graph G is a UDG if there is an assignment of unit disks centered at its vertices such that two vertices are adjacent if and only if one vertex is within the unit disk centered at the other vertex.

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.

Neighbourhood of a vertex: For each $v \in V$, the neighbourhood of *v* is defined by $N(v) = \{u \in V / u$ 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*.

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 $\gamma t(G)$.

Edge 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 edge graphs. One such formulation is the *Y*-dominating set of a edge graph.

Y-dominating set: Let $G' = (X, Y, E)$ be a edge 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 [9].

IV. Proposed Algorithm

A. *Edge Dominating Algorithm*

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

Algorithm 1: Total Edge Dominating (TED) set based algorithm for CDS

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

Output: Minimum connected edge dominating set $S \subseteq E$.

Initialization:

$S \leftarrow \phi$; $S' \leftarrow \phi$;

begin

- For the graph G(V,E), corresponding edge graph E(G) is constructed.
- $E(G) = (X, Y, E')$ where $X = E$, $Y = E'$ is a copy of E and $E' = \{(x, y')/(x, y) \in V\}$
- Partition the set S′ into subsets S1, S2, ..., Sk such that ∃ a edge between any two vertices in Si, i = 1, 2, ..., k
- $C\alpha \leftarrow$ any one of collection of $({}^{k}{}_{\alpha})$ Si, i =1, 2, ..., k
- Search for the common element $e \in \cap_{p \in C\alpha} B_p$ based on the sets of combination $p \in C\alpha$ - $S \leftarrow S \cup \{e\}$
- combine all the set elements in Cα together with the last set element of the combination and also add the vertex into the set. i.e., suppose if $e \in B1 \cap B2$ for one combination of the sets. S1S2 then $S2 = S2 \cup S1 \cup \{v\}$
- local search (V,E,G[S])

 ISSN: 2454-4248
213 – 219

end

Algorithm 2: Y − dom(X, Y, E′)
Input: Edge Graph E (G) of given graph G(V,E)
Output: y - dominating set E (G) = Total Edge dominating set of G(V,E)

Algorithm 3: local search (V, E ,G[S])
while S is connected, ∀ e ∈ S

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

In the first phase of the algorithm, for a given graph *G* the corresponding edge graph *E(G)* is constructed and its Y-dominating set is found. Procedure is described in Algorithm 2. By the Theorem 1, we get the total edge dominating set *E′* of a graph *G*. Then *E* is initialized as *E′*.

In the second phase, the TED algorithm partitions the set *E′* in to subsets $S1,S2,...,Sk$ where each Si's are connected. Then we search for a common element $e \in E - S$ such that $N(e) \cap Si \neq \phi$, $1 \leq i \leq k$. If such an element exists, then S is updated as S $\cup \{e\}$. Otherwise, repeat the above procedure for $(^{K}{}_{\alpha})$ combinations of Si's where $2 \leq \alpha \leq k - 1$. For any one of the combination in $(^{K}{}_{\alpha})$ say p, the above condition is satisfied then the corresponding element e $\in E - S$ is added into the set S. From the above if $S1,S2, ...,Sp$ along with e is connected, assigning Sp with $Sp \cup \{e\}$. Now, $Sp,Sp+1, ...,Sk$ forms a partition of the set S'. 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 Si and Sj using the criteria $N(Si) \cap N(Sj) = \phi$ where $i = 1$ to $k - 1$ and $j = i + 1$ to k. Then by neighbourhood search procedure technique, the TED algorithm finds two adjacent edges in $E - S$ such that one edge is adjacent to Si and the other edge is adjacent to Sj. By this way, all Si's are connected.

In the third phase, drop the redundant elements in the set *S* to get a MCEDS set using the exhaustive local search procedure in Algorithm 3.

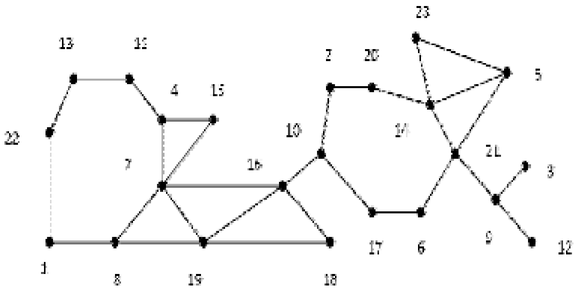

Figure 1: Initial topology of the network

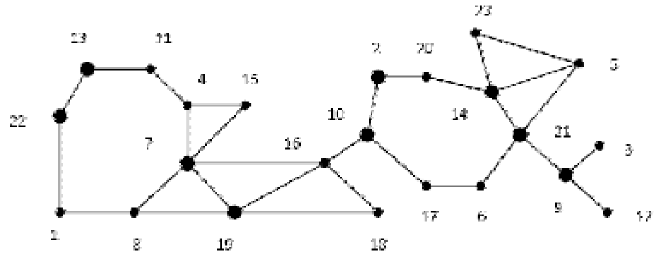

Figure 2: Total edge dominating set of the network

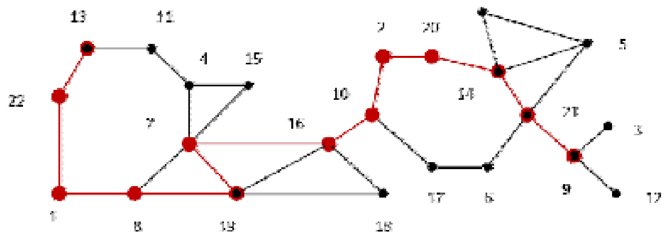

Figure 3: More edge connectors are selected

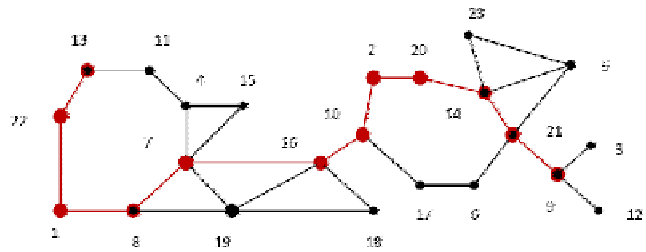

Figure 4: MCEDS of the network

Figure 1 shows the topology of network taken. In which the set of vertices in the graph below with larger radius represents the total edge dominating set. The total edge dominating set is partitioned into four subsets S1 = {7, 19}, S2 = {13, 22}, S3 = {9, 14, 21} and S4 = {2, 10}, which is shown in the Figure 2. Based on the TED algorithm procedure, now the vertex 16 is added to the total edge dominating set. Now, S2 = {13, 22}, S3 = {9, 14, 21} and S4 = {2, 7, 10, 16, 19}. Similarly the vertex 20 is added to the total edge dominating set, which is shown in the Figure 3. Further the vertex 19 is removed using the exhaustive local search procedure to get a MCDS set and final MCEDS set shown in the Figure 4.

Theorem 2 : The proposed TED algorithm returns a connected edge 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 edge dominating set *S′* and in the second phase of the algorithm the set *S′* partitioned into smaller subsets such that each subset is a connected set. 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 S and produces updated CDS S. Therefore it is enough to prove that updated S is still 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 $O(mn)$.

V. SIMULATION RESULTS

The performance of the algorithm constructing the CDS for networks, simulation is conducted. These simulations are

TABLE I. PARAMETERS FOR SIMULATION

Sim	**PARAMETERS**		
	n	***r***	***Area***
1	Varies	150m,200m	1000m X1000m
2	300, 600	Varies	1000m X 1000m

2 sets based on the parameters of a network such as number of nodes, transmission radio range and area density of the network and Table-1gives this information.

Simulations were implemented in MATLAB. To carry out the effectiveness of the TED algorithm, comparison made with the three recently developed algorithms of MCDS presented in [10, 11, 12] and are noted as WVA [10], JY [11] and CH [12]. In the simulation graphs, 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 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.

To evaluate the performance of the 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 are 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, Average size of CDS is taken as the size of the CDS produced by each algorithm and the obtained results are shown in the Figure 5. From the results shown 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 it is observed that the proposed TED algorithm find better average results than other compared algorithms.

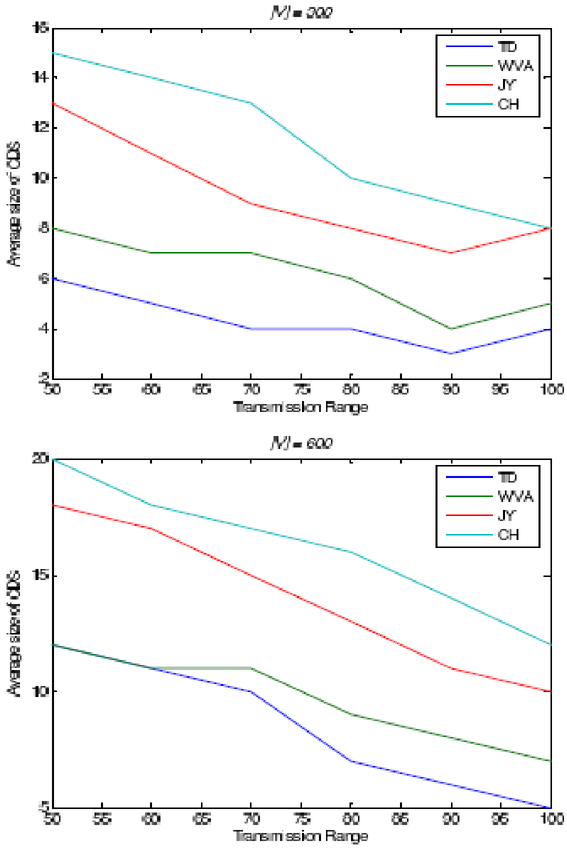

Figure 5: Comparison of average size for different transmission ranges.

In the simulation, graph compares the size of CDS of all the four algorithms when the transmission radio range varies. In simulation initially 300 nodes are randomly distributed into a fixed area of size 1000m × 1000m. Each node has been assigned to a transmission range starting from 50m, 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 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 Figure 5. the algorithms JY [11] and CH [12] deviated highly form the constructed size of the CDS when compared to WVA [10] and the TED algorithm.

Figure 6: Comparison of average size of CDS in different area densities, here graph shows the effectiveness of the TED 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.

ISSN: 2454-4248
213 – 219

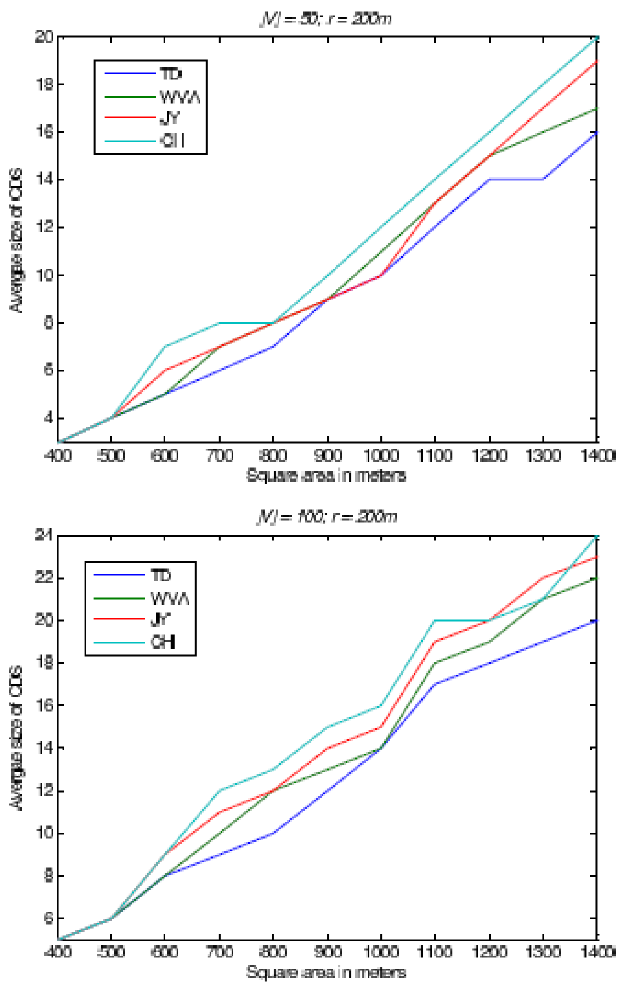

Figure 6: Comparison of average size for different densities.

VI. Conclusion

In this paper a novel approach called TED algorithm is established to construct the CDS, a virtual backbone of wireless networks, based on the total edge dominating set and domination theory of graphs. In the earlier research approaches of construction of connected dominating set, most of them based on the independent set construction, some complicated strategies are applied even though CDS is one
of the well known graph optimization problem. The proposed TED algorithm is purely based on the relation between the total edge dominating set, Y-dominating set of and the CDS. The total dominating set is best lower bound for the CDS than the dominating set, the proposed approach works on to reduce the computational complexity.

The conducted simulation on different important factors such as transmission ranges and area of network density reveals that the approach is better the recently developed approaches in the construction of CDS. The heuristics make them an attractive alternative approach for solving the graph optimization problems in dynamic environments.

VII. References

[1] S. Guha, S. Khuller, Approximation algorithms for connected dominating sets, Algorithmica 20 (1998) 374–387.

[2] Mathieu Couture, Michel Barbeau, Prosenjit Bose and Evangelos Kranakis ,“ Incremental Construction of k-Dominating Sets in Wireless Sensor Networks”, School of Computer Science Carleton University September 18 2006.

[3] Faisal N. Abu-Khzam, Christine Markarian, Friedhelm Meyer auf der Heide , Michael Schubert , “Approximation and Heuristic Algorithms for Computing Backbones in Asymmetric Ad-Hoc Networks”, arXiv:1510.01866v1 [cs.NI] 7 Oct 2015.

[4] Jiguo Yu, Nannan Wang, Guanghui Wang, Dongxiao Yu, “Connected dominating sets in wireless ad hoc and sensor networks – A comprehensive survey”, 0140-3664/$ - see front matter 2012 Elsevier B.V.All rights reserved. http://dx.doi.org/10.1016/j.comcom.2012.10.005.

[5] “A 5+ε-approximation algorithm for minimum weighted dominating set in unit disk graph”, Decheng Dai, Changyuan Yu, Institute for Theoretical Computer Science, Tsinghua University, Beijing, 100084, PR China

[6] Ning Chen, Jie Meng, Jiawei Rong, Hong Zhu, “Approximation For Dominating Set Problem With Measure Functions”, Department of Computer Science, Fudan University, P. R. China, Computing and Informatics, Vol. 23, 2004, 37–49

[7] Samir Khullerz, Sudipto Guhay ,“Approximation Algorithms for Connected Dominating Sets”, , Dept. of Computer Science University of Maryland,College Park, MD 20742, Dept. of Computer Science and UMIACS, University of Maryland,College Park, MD 20742.

[8] Khaled M. Alzoubi Peng-Jun Wan Ophir Frieder “New Distributed Algorithm for Connected Dominating Set in Wireless Ad Hoc Networks”, Department of Computer Science Illinois Institute of Technology Chicago, IL 60616, Proceedings of the 35th Hawaii International Conference on System Sciences - 2002

[9] Balaji, K. Kannan, Y. B. Venkatakrishnan, “Total dominating set based algorithm for connected dominating set in Ad hoc wireless networks” WSEAS TRANSACTIONS on MATHEMATICS S. E-ISSN: 2224-2880 1166 Issue 12, Volume 12, December 2013

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

[11] 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.v

[12] J. Yu, N. Wang, G. Wang, Heuristic Algorithmsvfor 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.