CONSTRUCTION OF STRATEGIC CONNECTED DOMINATING SET FOR MOBILE AD HOC NETWORKS

Victor Ceronmani Sharmila and Amalanathan George

Department of Information Technology, Hindustan University, Chennai, India

ABSTRACT

Connected Dominating Sets (CDSs) can serve as virtual backbones for wireless networks. A smaller virtual backbone (minimum size CDS) incurs less communication overhead. Unfortunately, computing a minimum size CDS is NP-hard and thus many algorithms were developed which concentrates on how to construct smaller CDSs. Aiming for minimum number of nodes in CDS, results in CDSs that are highly unstable. Here we present four CDS algorithms which are constructed based on strategy, density, Threshold Neighborhood Distance Ratio (TNDR) and velocity. Any node in the network can be a strategic node depending upon the application and need and will be selected as the starting node instead of the node with highest density which concentrates on minimum size CDS. Simulation methods are used to establish the efficiency of the proposed algorithm.

1. INTRODUCTION

Wireless networks, like Mobile Ad hoc Network (MANET), is a dynamic distributed system of arbitrarily moving wireless nodes. They are easy to deploy to an application field as the cost of creating a network is relatively low. However the shortcomings of MANET also limit the network performance. The wireless nodes have limited energy, which affects the life time of the network. All network functions, such as determining the network topology, multiple access and routing of data over the most appropriate multi-hop paths, must be performed in a distributed way, as there is no access point to have a centralized control. These tasks are particularly challenging due to limited communication bandwidth available in wireless channel. Wireless nodes possess limited transmission range. If any two nodes are within the maximum transmission range of each other, they can exchange messages directly. But fixing the transmission range for a wireless node is critical. If the transmission range is limited then network gets partitioned, as a result, multi-hop communication is very common in MANETs. If the transmission range is large enough, the number of nodes that locally compete over a shared channel increases.

Wireless networks have no predefined physical backbone infrastructure for topology control, which makes routing-related tasks more complicated. This can be overcome by employing a virtual backbone in a wireless network. A virtual backbone network has several benefits in routing and managing a wireless network. For instance the concept of virtual backbone network can be applied in MANETs to minimize the message overhead during broadcast communication. Traditionally broadcast communication is accomplished through flooding in which each wireless node receives a copy of the broadcast message from all of its neighbors and is also responsible for forwarding the message exactly once to all of its neighbors. Flooding is often considered to trigger the "broadcast storm problem" as it is associated with redundant retransmission of messages. The virtual backbone network is a structure that is similar to the physical backbone infrastructure of packet cellular networks.

Corresponding Author: Victor Ceronmani Sharmila, Department of Information Technology, Hindustan University, Chennai, India

First approximation algorithm to compute a minimum size virtual backbone using a Connected Dominating Set (CDS) in Unit Disk Graphs (UDGs) was introduced by Guha and Kuller (1998), which becomes one of the major approaches to construct a virtual backbone for a given graph representing a wireless network. Mobile ad hoc networks are often represented as a unit disk graph in which vertices represent wireless node and a bidirectional edge represents wireless connection. A Dominating Set (DS) is a subset of nodes of a network such that every node that is not in the DS is directly connected to at least one member of DS. A Connected Dominating Set (CDS) is defined as a set of nodes in a network such that each node is either in the set or one hop to a node in the set. In addition, every node in a CDS should be able to reach every other node in the CDS by a path that stays entirely within the CDS.

Previous works (Sakai *et al*., 2008; Sheu *et al.*, 2009), demonstrate the use of CDS based virtual backbones to propagate the broadcast messages so that they are exchanged only among the nodes in the CDS instead of being broadcast by all the nodes in the network, thus reducing the number of unnecessary retransmissions (Fly and Meghanathan, 2010). A minimum connected dominating set is the smallest CDS (in terms of number of nodes in the CDS) for the entire graph. For a virtual backbone-based broadcast communication, the smaller a CDS, the efficiency of its performance is better. Since each node in a wireless network shares its communication channel with its neighbors, a smaller virtual backbone suffers less from the interference problem. Additionally, it also makes the maintenance of the virtual backbone easier. For this reason, most research studies in this area focus on reducing the size of a CDS (Thai *et al*., 2007a; 2007b; Funke *et al*., 2006; Zeng *et al*., 2006). Unfortunately, the problem of determining the minimum connected dominating set in an undirected graph is NP-complete.

The efficiency of broadcasting depends on the CDS node size that directly influences the number of redundant retransmissions. If the size of the CDS is too large, maintenance and searching in time will become difficult. If the size of the CDS is too small, some characteristics in original networks may be lost. But aiming for smaller sized CDSs in MANETs results in CDSs that are highly unstable. The CDS itself has to be frequently rediscovered and this adds considerable

overhead in the lifetime of a CDS. Considering only one parameter, such as, density is not sufficient to increase the life time of the constructed CDS. Therefore the parameters such as strategy, TNDR ratio and velocity should also be considered while constructing the CDS. In previous works, the starting node of the CDS is maximum density based (Meghanathan, 2012a), minimum velocity based (Meghanathan, 2010), identity based (Velummylum and Meghanathan, 2010), strong neighborhood based (Meghanathan and Terrell, 2012), node stability index based (Meghanathan, 2012b) and predicted link expiration time based (Fly and Meghanathan, 2010).

In this study we propose four strategic algorithms for construing the CDS. The algorithms are Strategic Maximum Density (ST-MaxD) CDS, Strategic Strong Neighborhood (ST-SN) CDS, Strategic Minimum Velocity Minimum Density (ST-MV-MD) CDS and Strategic Minimum Density Minimum Velocity (ST-MD-MV) CDS. Here the CDS will be having a strategic node as the starting node; any node in the network can be selected as a strategic ode based on the application and need. For example it can be a source, DNS, web proxy, internet provider, gateway or door way node, beacon, security key provider, repeater, rescuer, router. Once a node is added to the CDS, all its neighbors are said to be covered. Then the algorithm prefers to include the nodes which satisfy the specific criteria.

The specific criterion of ST-MaxD CDS algorithm is to include the nodes in the decreasing order of the number of uncovered neighbors until all nodes in the network are covered. The ST-SN CDS algorithm is constructed by considering an input parameter called the TNDR. For an edge to be part of a strong neighborhood-based topology, the ratio of the physical Euclidean distance between the end nodes of the edge to that of the transmission range per node (edge weight) has to be less than or equal to the TNDR. After pruning the edges whose edge weight is greater than TNDR the neighbors of the strategic node is said to be covered. The specific criterion of ST-SN CDS algorithm is also to include the nodes in the decreasing order of the number of uncovered neighbors until all nodes in the network are covered.

The specific criterion of ST-MV-MD CDS algorithm is to include the nodes in the increasing order of their velocity until all nodes in the network are covered. If there is a tie then the node with minimum number of uncovered neighbors is chosen. The specific criterion of ST-MD-MV CDS algorithm is to include the nodes in the increasing order of the number of uncovered

neighbors until all nodes in the network are covered. If there is a tie then the node with minimum velocity is chosen. If still there is a tie then the node is chosen randomly. Further the algorithm also checks whether the selected node forms circuits between CDS nodes, as circuits increase the count of retransmission messages. Our contribution in this study is to construct a CDS with strategy, longer lifetime and minimum communication overhead.

2. DEFINITIONS

- *CDS-Node-List*: Nodes that are members of the CDS.
- *CDS-Edge-List*: Edges that exist between any two CDS nodes.
- *Covered-Nodes-List*: Nodes that are in the *CDS-Node-List* and all nodes that are one hop to at least one member of the *CDS-Node-List*.
- *CDS-Circuit-List*: Circuits that are formed by the nodes from the *CDS-Node-List*.
- Density: A node's maximum number of uncovered neighbors.
- Homogenous network: A network in which all nodes function under the same fixed transmission range R.
- Open neighbourhood of a node i (ON_i): The set of neighbor nodes such that the physical Euclidean distance between node i and each node j in this set is $\leq R$. Every node j $\in ON_i$ is simply referred to as a "neighbor" of node i.
- Threshold Neighbor Distance Ratio (TNDR): The maximum value of the ratio of the physical Euclidean distance between a node i and a node j in its open neighborhood and the transmission range R.
- Strong Neighborhood of a node i (SN_i): The subset of nodes in its open neighborhood, such that the ratio of the physical Euclidean distance between node i and any node j in SN_i is less than or equal to the TNDR.
- Edge Weight (EW): The ratio of physical Euclidean distance between the nodes constituting the edge to the fixed transmission range R of the nodes.
- Node Stabiltiy Index (NSI): The sum of predicted expiration times of the links (LETs) with the neighbor nodes that are not yet covered by a CDS node.
- Static Network Graph: A snapshot of the network topology at a particular time instant.

3. REVIEW OF RELATED WORKS

Even though many algorithms are available to identify a CDS, almost all the algorithms follows the general criteria for CDS node selection, which is: the node cannot already be a part of *CDS-Node-List*, the node must be in the *Covered-Nodes-List* and the node must have at least one uncovered neighbour. The nodes which are adjacent to the selected node are covered. The next node to be part of CDS is selected from the *Covered-Nodes-List*. This process is repeated until all the nodes in the network are covered.

Velummylum and Meghanathan (2010), the authors have proposed an identity based CDS (ID CDS) algorithm according to which the nodes with larger IDs are preferred for inclusion into the CDS. Meghanathan (2010), the author had proposed a minimum velocity-based CDS (MinV CDS) algorithm according to which slow moving nodes with lower velocity are preferred for inclusion into the CDS. Fly and Meghanathan (2010), the authors have proposed a predicted link expiration time based CDS (LET CDS) algorithm according to which, the edge with largest predicted link expiration time is preferred for inclusion into the CDS. Once the edge is included into the *CDS-Edge-List*, the two constituent nodes of the edge are included into the *CDS-Node-List*.

Meghanathan (2012b), the author has proposed a node stability index based CDS (NSI CDS) algorithm according to which the nodes with larger Node Stability Index (NSI) are preferred for inclusion into the CDS. Meghanathan and Terrell (2012), the authors have proposed a strong neighbourhood based CDS (SN CDS) algorithm according to which the nodes with maximum number of uncovered neighbors are preferred for inclusion into the CDS. This algorithm considers an input parameter called the Threshold Neighborhood Distance Ratio (TNDR), for an edge to be part of a strong neighborhood-based topology.

4. ALGORITHMS TO CONSTRUCT STRATEGIC CONNECTED DOMINATING SET (ST-CDS)

We present the ST-MaxD CDS, ST-SN CDS, ST-MV-MD CDS and ST-MD-MV CDS algorithms to finalize the nodes into the CDS. The input to all the algorithms is a snapshot of the network as shown in **Fig. 1a**. In that nodes are symbolized as circles and the numerical number inside the circle represents the density of the node and the numerical number outside the circle represents the ID of the node.

In the following figures, nodes which are symbolized as dotted circles with a plus symbol are qualified to be part of *CDS-Node-List*, the edges between any two CDS nodes in the *CDS-Node-List* are qualified to be a part of *CDS-Edge-List* and the nodes which are represented as dotted circles are qualified to be a part of *Covered-Nodes-List*.

The edge weight is represented as real number. **Table 1** is created for representing the values of identity, density and velocity for the initial static graph with 24 nodes and 44 edges.

The general steps for ST-MaxD CDS and ST-SN CDS algorithms are:

- Any node in the network which is strategically important can be selected as the start node to be included into the *CDS-Node-List* and all its one hop neighbors are said to be covered and they are included into the *Covered-Nodes-List* along with the start node. Then
    - The density of the start node is changed to 0
    - The density of covered nodes is computed
- The next node is chosen from the *Covered-Nodes-List* which satisfies the following conditions
    - The node should not be present in *CDS-Node-List*
    - The node should be present in *Covered-Nodes-List*
    - The node must have at least one uncovered neighbour
    - The node must satisfy the specific criterion
    - If there is a tie, choose arbitrarily
- If a node is chosen from step 2, then it is added to the *CDS-Node-List* and *Covered-Nodes-List* and its entire one hop neighbors are added to the *Covered-Nodes-List*
- The density of the selected node is changed to 0 and density of all other nodes (except for CDS nodes and the nodes whose density is 0) is recomputed again
- The above process is continued until there are no uncovered neighbors in the network
- The *CDS-Node-List* an*d CDS-Edge-List* are computed

4.1. Strategic Maximum Density CDS (ST-MaxD CDS)

The specific criterion of this algorithm is to select a node with maximum number of uncovered neighbors. This algorithm is illustrated below with an example.

Any node in the network can be selected as the starting node. In this example node 7 is chosen which has density 5 as shown in **Fig. 1b** and it is included in the *CDS-Node-List* and *Covered-Nodes-List*. Then node 7's covered neighbors (node 3, 4, 8, 11, 12) are added into the *Covered-Nods-List*. The density of the start node (node 7) is changed to 0 and the density of all other nodes is computed as shown in **Fig. 1b**.

Then the node which has highest density is chosen from the *Covered-Nodes-List* as shown in **Fig. 1c** and the process is continued until all the nodes are covered as shown in **Fig. 1c and 1d**. The number of CDS nodes in *CDS-Node-List* is 11 and the number of edges in *CDS-Edge-List* is 15 as shown in **Fig. 1e**. There are five circuits in **Fig. 1f**, so *CDS-Circuit-List* = 5 and members of the circuit = {11, 15, 16}, {15, 16, 20}, {10, 14, 19, 15}, {15, 19, 20} and {14, 18, 19}.

4.2. Strategic Strong Neighborhood CDS (ST-SN CDS)

The specific criterion of this algorithm is to prune the edges of the initial static network whose TNDR value is greater than 0.5. Then select a node with maximum number of uncovered neighbors. This algorithm is illustrated below with an example.

The strong neighborhood network is formed from initial static network graph (**Fig. 1a**) after pruning the edges which as edge weight greater than 0.5 (**Fig. 2a**). Any node in the strong neighbourhood network can be selected as the starting node.

Table 1. Parameter values

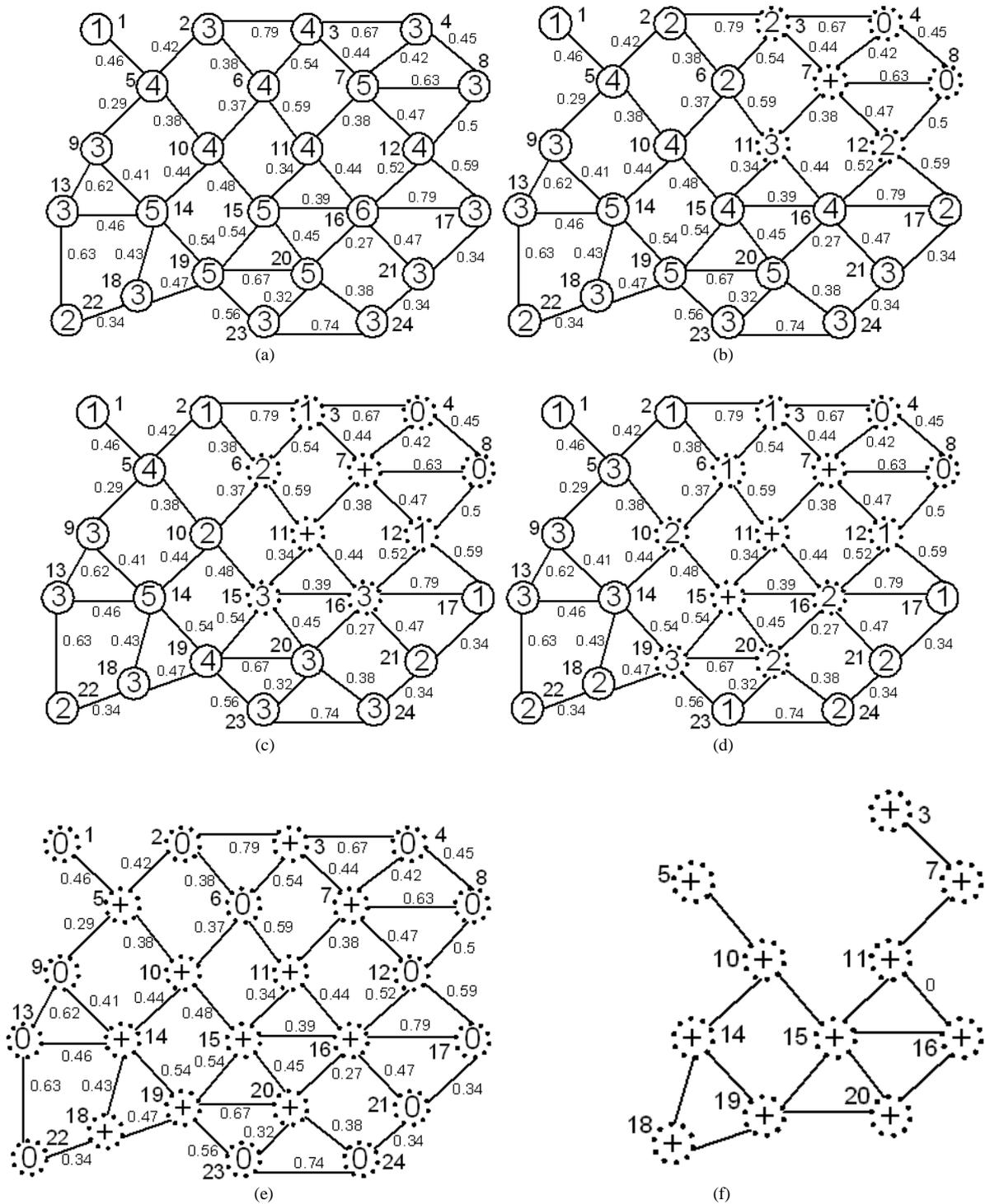| Identity	Density	Velocity
1	1	1.2
2	3	0.6
3	4	1.9
4	3	6.3
5	4	3.2
6	4	7.5
7	5	6.7
8	3	4.5
9	3	7.0
10	4	9.8
11	4	3.8
12	4	3.9
13	3	7.6
14	5	7.4
15	5	1.5
16	6	5.3
17	3	0.2
18	3	7.7
19	5	6.9
20	5	4.0
21	3	9.8
22	2	7.6
23	3	8.4
24	3	9.9

Fig. 1. Example to construct ST-MaxD CDS (a) Initial static network graph, (b) Iteration # 1 (c) Iteration # 2, (d) Iteration # 3, (e) Iteration # 11(Last Iteration) (f) CDS Nodes, edges and circuits list
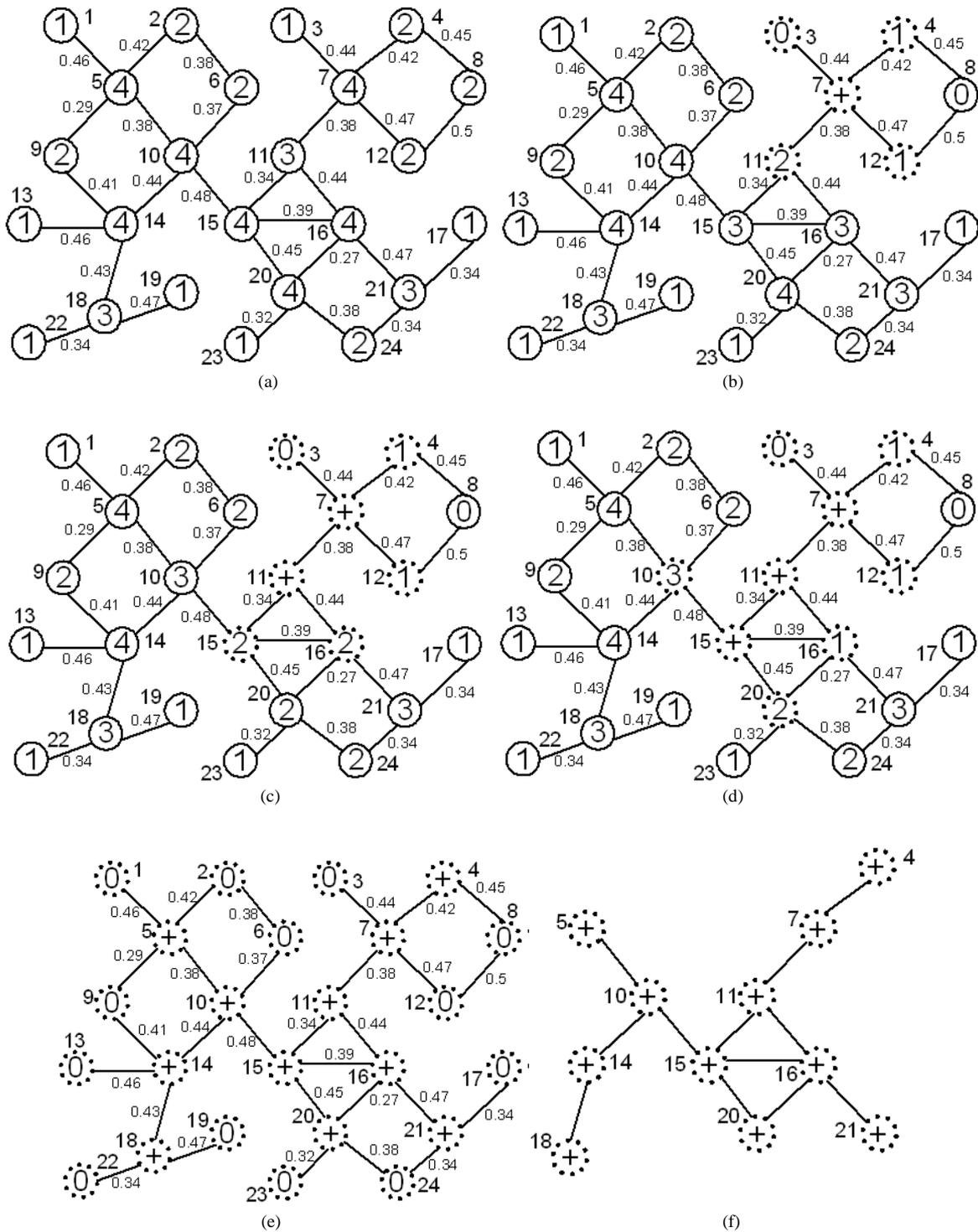
Fig. 2. Example to construct ST-SN CDS, (a) Strong neighborhood network graph, (b) Iteration # 1, (c) Iteration # 2, (d) Iteration # 3, (e) Iteration # 11(Last Iteration), (f) CDS Nodes, edges and circuits
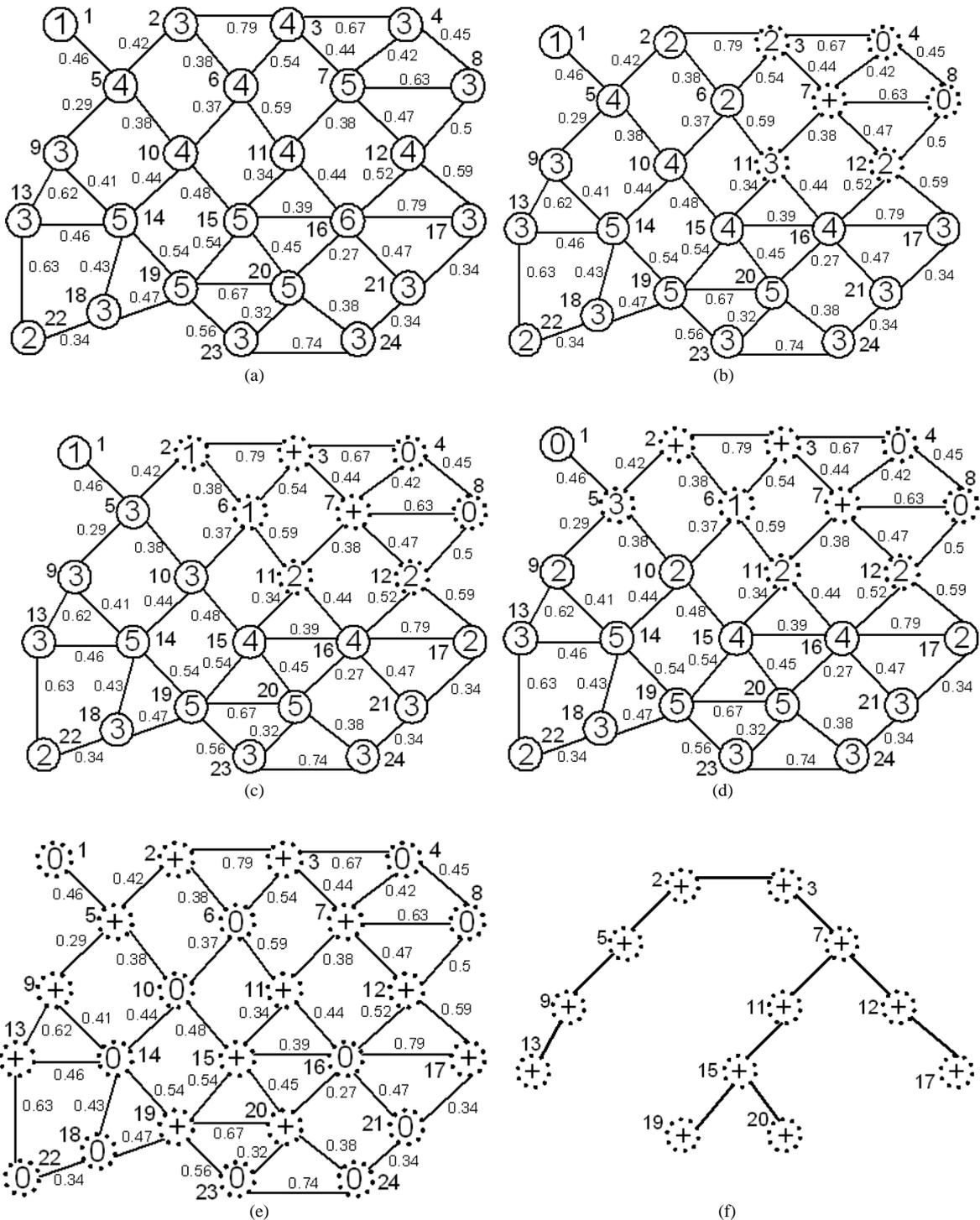
Fig. 3. Example to construct ST-MV-MD CDS, (a) Initial static network graph, (b) Iteration # 1, (c) Iteration # 2, (d) Iteration # 3, (e) Iteration # 12(Last Iteration), (f) CDS nodes, edges and circuits

Fig. 4. Example to construct ST-MD-MV CDS, (a) Initial static network graph, (b) Iteration # 1, (c) Iteration # 2, (d) Iteration # 3, (e) Iteration # 12(Last Iteration), (f) CDS nodes, edges and circuits
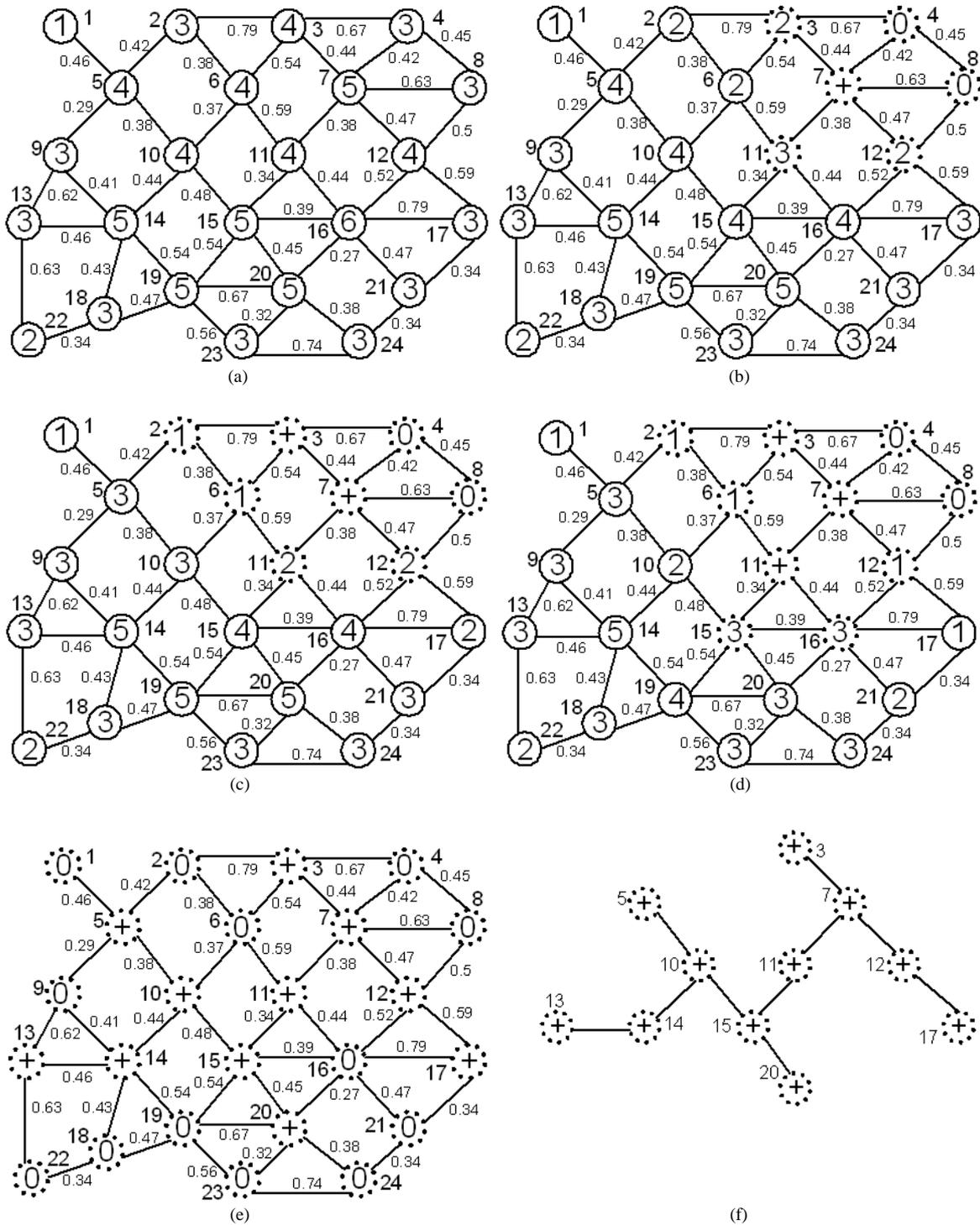
In this example node 7 is chosen which has density 4 as shown in **Fig. 2b** and it is included in the *CDS-Node-List* and *Covered-Nodes-List*. Then node 7's covered neighbors (node 3, 4, 11, 12) are added into the *Covered-Nods-List*. The density of the start node (node 7) is changed to 0 and the density of all other nodes is computed as shown in **Fig. 2b**.

Then the node which has highest density is chosen from the *Covered-Nodes-List* as shown in **Fig. 2c** and the process is continued until all the nodes are covered as shown in **Fig. 2c and 2d**. The number of CDS nodes in *CDS-Node-List* is 11 and the number of edges in *CDS-Edge-List* is 12 as shown in **Fig. 2e**. There are two circuits in **Fig. 2f** so *CDS-Circuit-List* = 2 and members of the circuit = {11, 15, 16} and {15, 16, 20}.

4.3. Strategic Minimum Velocity Minimum Density CDS (ST-MV-MD CDS)

The specific criterion of this algorithm is to select a node with minimum velocity. If there is a tie choose a node which has minimum number of uncovered neighbors (greater than one). This algorithm is illustrated below with an example.

Any node in the network can be selected as the starting node. In this example node 7 is chosen which has density 5 as shown in **Fig. 3a** and it is included in the *CDS-Node-List* and *Covered-Nodes-List*. Then node 7's covered neighbors (node 3, 4, 8, 11, 12) are added into the *Covered-Nods-List*. The density of the start node (node 7) is changed to 0 and the density of all other nodes is computed as shown in **Fig. 3b**.

Then the node which has minimum velocity is chosen from the *Covered-Nodes-List* as shown in **Fig. 3b** and the process is continued until all the nodes are covered as shown in **Fig. 3c-e**. Whenever there is a circuit, a suitable edge can be removed so that to make it a tree, usually edges from a vertex with lower ID will not be removed. The number of CDS nodes in *CDS-Node-List* is 12 and the number of edges in *CDS-Edge-List* is 11 as shown in **Fig. 3f**.

4.4. Strategic Minimum Density Minimum Velocity CDS (ST-MD-MV CDS)

The specific criterion of this algorithm is to select a node with minimum density (greater than one). If there is a tie or the density of all the nodes present in the *Covered-Nodes-List* is less than two, choose a node with minimum velocity. This algorithm is illustrated below with an example.

Any node in the strong neighbourhood network can be selected as the starting node. In this example node 7 is chosen which has density 5 as shown in **Fig. 4a** and it is

included in the *CDS-Node-List* and *Covered-Nodes-List*. Then node 7's covered neighbors (node 3, 4, 8, 11, 12) are added into the *Covered-Nods-List*. The density of the start node (node 7) is changed to 0 and the density of all other nodes is computed as shown in **Fig. 4b**.

Then the node which has minimum density (greater than one) is chosen from the *Covered-Nodes-List* as shown in **Fig. 4b** and the process is continued until all the nodes are covered as shown in **Fig. 4c-e**. Whenever there is a circuit, a suitable edge can be removed so that to make it a tree, usually edges from a vertex with lower ID will not be removed. The number of CDS nodes in *CDS-Node-List* is 11 and the number of edges in *CDS-Edge-List* is 10 as shown in **Fig. 4f**.

5. SIMULAIONS AND ANALYSIS

The proposed solution was implemented on an ad hoc peer-peer network by placing around 150 nodes in an M×M area. Nodes were placed randomly and the proposed algorithms were performed on a network with varying number of nodes and density. The proposed four algorithms and existing four algorithms were simulated in Visual C++ Express tool. Networks were generated with sizes of 25, 50, 100 and 150 respectively. We use the Breadth First Search (BFS) algorithm on the *CDS-Node-List* and *CDS-Edge-List* and examine whether the underlying CDS is connected or not. The BFS algorithm checks whether all the nodes in the CDS are connected. If at least one node is not connected then the algorithm fails, which initiates the necessity for constructing a new CDS. The complexity of ST-MaxD CDS and ST-SN CDS algorithm are $O(n^2)$ and the complexity of ST-MV-MD CDS and ST-MD-MV CDS algorithm are $O(n^4)$.

We evaluate the CDS algorithms with respect to the three metrics: CDS node size, CDS edge size and CDS circuit size where CDS node size represents the number of nodes present in the *CDS-Node-List*, CDS edge size represents the number of edges present in the *CDS-Edge-List* and CDS circuit size represents the number of circuits present in the *CDS-Circuit-List*. The number of circuits formed and other parameters are shown in **Table 2**, where n represents number of nodes; e represents number of edges; cn represents number of CDS nodes; ce represents number of CDS edges; c represents number of circuits. The smallest dominating set for each network was recorded. The comparison of four proposed algorithms with respect to CDS node list and CDS edge list is shown in **Fig. 5a and 5b**.
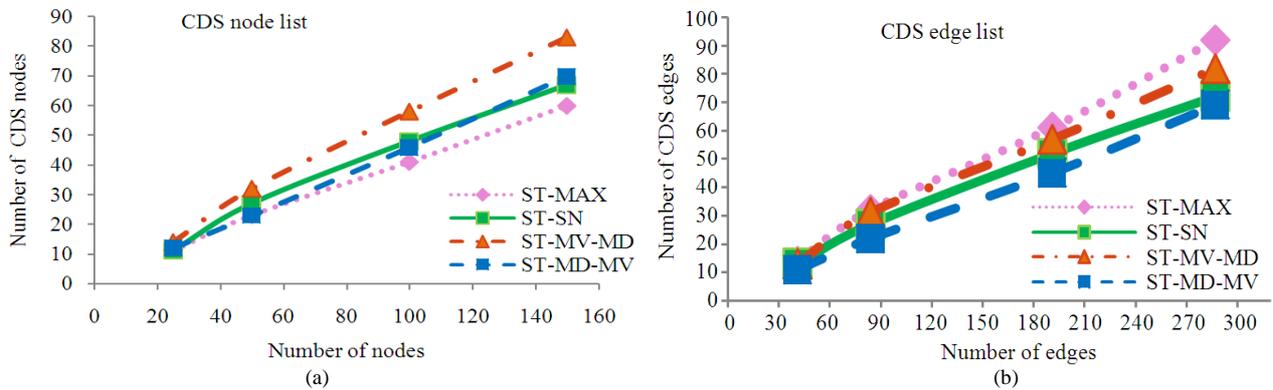
Fig. 5. Comparison of four algorithms, (a) comparison of CDS node list, (b) comparison of CDS edge list

Table 2. Estimation of parameters

Initial static network		MaxD CDS			SN CDS		
n	e	cn	ce	c	cn	ce	c
25	41	11	15	1	11	14	2
49	84	23	33	3	27	34	3
100	191	42	65	10	49	54	7
144	287	60	98	13	68	75	10
		MinV CDS			ID CDS		
25	41	15	22	4	13	17	2
49	84	35	49	7	30	38	5
100	191	58	81	15	49	76	12
144	287	76	116	19	69	106	15
		ST-MaxD CDS			ST-SN CDS		
25	41	11	14	2	11	13	1
49	84	23	32	3	27	32	2
100	191	41	61	8	48	52	5
144	287	60	92	12	67	72	8
		ST-MV-MD CDS			ST-MD-MV CDS		
25	41	14	13	0	12	11	0
49	84	32	31	0	23	22	0
100	191	58	57	0	46	45	0
144	287	83	82	0	70	69	0

6. CONCLUSION

In this study, we have investigated the problem of constructing a strategic and energy efficient CDS for a wireless network. We introduced four CDS algorithms to study the performance metrics CDS node size, CDS edge size and CDS circuit size and compared with the previous algorithms such as MaxD CDS, SN CDS, MinV CDS and ID CDS. The ST-MaxD CDS is having less number of CDS nodes and more number of edges and circuits. The ST-SN CDS is having moderate number of nodes, edges and circuits. Hence ST-MaxD CDS and ST-SN CDS algorithms can be used where speed of the network matters. The ST-MV-MD CDS is having more number of CDS nodes, moderate number of CDS edges and no circuits. The ST-MD-MV CDS is having moderate number of CDS nodes, less number of CDS edges and no circuits. Hence ST-MV-MC CDS and ST-MD-MV CDS can be used when it is necessary to decide networks with minimum energy. The results demonstrate that our algorithm outperform the other algorithms. In future authors may explore the possibiliteis for different kinds of networks using the proposed algorithms.

7. REFERENCES

Fly, P. and N. Meghanathan, 2010. Predicted link expiration time based connected dominating sets for mobile ad hoc networks. Comput. Sci. Eng., 0: 2096-2103.

Funke, S., A. Kesselman, U. Meyer and M. Segal, 2006. A simple improved distributed algorithm for minimum CDS in unit disk graphs. ACM Trans. Sensor Netw., 1: 444-453. DOI: 10.1145/1167935.1167941

Guha, S. and S. Khuller, 1998. Approximation algorithms for connected dominating set. Algorithmica, 1: 374-387. DOI: 10.1007/PL00009201

Meghanathan, N. and M. Terrell, 2012. An algorithm to determine stable connected dominating sets for mobile ad hoc networks using strong neighborhoods. Int. J. Combinat. Optim. Problems Inform., 1: 79-92.

Meghanathan, N., 2010. Use of minimum node velocity based stable connected dominating sets for mobile ad hoc networks. Comput. Appli. Special MANETs, 1: 89-96. DOI: 10.5120/1016-52

Meghanathan, N., 2012a. Applications of graph theory algorithms in mobile ad hoc networks. mobile computing techniques in emerging markets. Syst. Applic. Serv., 1: 98-125. DOI: 10.4018/978-1-4666-0080-5.ch004

Meghanathan, N., 2012b. A node stability index-based connected dominating set algorithm for mobile ad hoc networks. Netw. Commun., 1: 31-40. DOI: 10.1007/978-3-642-27299-8_4

Sakai, K., M.T. Sun and W.S. Ku, 2008. Maintaining CDS in mobile ad hoc networks. Wireless Algorithms Syst. Applic., 1: 141-153. DOI: 10.1007/978-3-540-88582-5_16

Sheu, P.R., H.Y. Tsai, Y.P. Lee and J.Y. Cheng, 2009. On calculating stable connected dominating sets based on link stability for mobile ad hoc networks. Tamkang J. Sci. Eng., 1: 417-428.

Thai, M.T., F. Wang, D. Liu, S. Zhu and D.Z. Du, 2007a. Connected dominating sets in wireless networks with different transmission ranges. IEEE Trans. Mobile Comput., 1: 721-730. DOI: 10.1109/TMC.2007.1034

Thai, M.T., N. Zhang, R. Tiwari and X. Xu, 2007b. On approximation algorithms of k-connected m-dominating sets in disk graphs. Theoretical Comput. Sci., 1: 49-59. DOI: 10.1016/j.tcs.2007.05.025

Velummylum, N. and N. Meghanathan, 2010. On the utilization of id-based connected dominating sets for mobile ad hoc networks. Int. J. Adv. Res. Comput. Sci., 1: 36-43.

Zeng, Y., X. Jia and H. Yanxiang, 2006. Energy efficient distributed connected dominating sets construction in wireless sensor networks. Proceedings of International Conference on Wireless Communications and Mobile Computing, Jul. 03-06, ACM Press, New York, USA., pp: 797-802. DOI: 10.1145/1143549.1143709