

A GENERIC ALGORITHM TO DETERMINE CONNECTED DOMINATING SETS FOR MOBILE AD HOC NETWORKS AND PERFORMANCE EVALUATION UNDER DIFFERENT MOBILITY MODELS

Natarajan Meghanathan and Ilin S. Dasari

Jackson State University
Jackson, MS, USA

Corresponding Author Email: natarajan.meghanathan@jsums.edu

ABSTRACT

The high-level contributions of this paper are: (1) A generic algorithm to determine connected dominating sets (CDS) for mobile ad hoc networks and its use to find CDS of different categories: maximum density-based (MaxD-CDS), node ID-based (ID-CDS) and stability-based (minimum velocity-based, MinV-CDS); (2) Performance comparison of the above three categories of CDS algorithms with respect to two categories of mobility models: random node mobility models (Random Waypoint model) and the grid-based vehicular ad hoc network (VANET) mobility models (City Section and Manhattan mobility models), with respect to the CDS Node Size and Lifetime. The performance of a CDS is observed to depend on the criteria used to form the CDS (i.e., the node selection criteria of the underlying algorithm) and the mobility model driving the topological changes. For each category of CDS, we identify the mobility model under which one can simultaneously maximize the lifetime and node size, with minimal tradeoff. For the two VANET mobility models, we also evaluate the impact of the grid block length on the CDS lifetime and node size.

KEYWORDS

Connected Dominating Sets, Mobile Ad hoc Networks, Algorithms, Mobility Models, Stability, Simulations

1. INTRODUCTION

A Mobile ad hoc network (MANET) is a category of wireless networks in which the topology changes dynamically with time due to arbitrary node movement. MANET routes are often multi-hop in nature (due to limited transmission range per node) and also temporally change depending on node mobility and availability. MANET routing protocols have been predominantly designed

to be on-demand in nature rather than using a proactive strategy of determining the routes irrespective of the need [1][2]. MANET on-demand routing protocols typically use a network-wide broadcast route-reply cycle (called flooding) to discover the appropriate communication topology (path, tree or mesh) [2]. A source node initiates the flooding of the Route-Request (RREQ) packets that propagate through several paths and reach the targeted destination (for unicasting) or receiver nodes (for multicasting). These nodes choose the path that best satisfies the principles of the routing protocol in use and respond with a Route-Reply (RREP) packet to the source on the selected route. With flooding, every node in the network is required to broadcast the RREQ packet exactly once in its neighborhood. Nevertheless, the redundancy of retransmissions and the resulting control overhead incurred with flooding is still too high, as every node spends energy and bandwidth to receive the RREQ packet from each of its neighbors.

A Connected Dominating Set (CDS) of a network graph comprises of a subset of the nodes such that every node in the graph is either in the CDS or is a neighbor of a node in the CDS. Recent studies (e.g. [3][4]) have shown the CDS to be a viable backbone for network-wide broadcasting of a message (such as the RREQ message), initiated from one node. The message is broadcast only by the CDS nodes (nodes constituting the CDS) and the non-CDS nodes (who are neighbors of the CDS nodes) merely receive the message, once from each of their neighboring CDS nodes. The efficiency of broadcasting depends on the CDS Node Size that directly influences the number of redundant retransmissions. One category of CDS algorithms for MANETs aim to determine CDSs with reduced Node Size (i.e., constituent nodes) such that whole network could be covered with as fewer nodes as possible. The other two categories of CDS algorithms are those based on node ids and those based on stability. The node-id based algorithm prefers to include nodes with lower id into the CDS until all the nodes in the network are covered. The stability-based category of CDS algorithms focus on maximizing the CDS Lifetime such that the control overhead incurred in frequently reconfiguring a CDS is reduced.

In this paper, we first present a generic algorithm that can be used to obtain a CDS under each of the above three categories. We pick the maximum density-based CDS (MaxD-CDS) [5] and ID-based CDS (ID-CDS) [6] to be representatives of the minimum node size and node id-based CDS algorithms respectively; the minimum velocity-based CDS (MinV-CDS) [7] algorithm is chosen to be the representative for the stability-driven CDS. The second half of the paper evaluates the three CDS algorithms under diverse conditions of network density and node mobility with respect to two critical metrics: CDS Lifetime and CDS Node Size, and under three different mobility models. The mobility models considered are the Random Waypoint model [8] that is often used to simulate the mobility of nodes in classical MANETs; and the City Section [9] and Manhattan [9] mobility models that are used to simulate the mobility of nodes in vehicular ad hoc networks (VANETs). This is the first such comprehensive comparison study of the CDS algorithms for ad hoc networks under different mobility models. The paper also identifies any potential CDS lifetime-node size tradeoff existing for the three categories of CDS algorithms.

The rest of the paper is organized as follows: Section 2 presents a generic CDS algorithm that can be used to study each of the three CDS algorithms (MaxD-CDS, ID-CDS and MinV-CDS) that will be analyzed in the simulations. Section 3 outlines the simulation environment, discusses the mobility models and presents the simulation results observed for the three CDS algorithms under the three mobility models. Section 4 presents the conclusions by outlining the significant observations and contributions of this paper. Throughout the report, the terms ‘vertex’ and ‘node’, ‘edge’ and ‘link’ are used interchangeably. They mean the same.

2. GENERIC ALGORITHM TO CONSTRUCT THE DIFFERENT CONNECTED DOMINATING SETS

In this section, we present a generic algorithm that can be used to construct the four different CDSs, with appropriate changes in the data structure (Priority-Queue) used to store the list of covered nodes that could be the candidate CDS nodes and the use of the appropriate criteria in each of the iterations of the algorithm to evolve the particular CDS of interest. The advantage with proposing a generic algorithm is that the different CDSs can be constructed with virtually the same time complexity.

Network Model: We assume a homogeneous network of wireless nodes, each operating at a fixed transmission range, R . We use the unit-disk graph model [10] according to which there exists a link between any two nodes in the network as long as the Euclidean distance is within the transmission range per node. The set of neighbors of a node i , $\text{Neighbors}(i)$, comprises of nodes that are connected to vertex i in the unit-disk graph model. A node learns about its own location through location service schemes such as the Global Positioning System (GPS) [11]. A node learns the location and mobility parameters (velocity and direction of movement – measured as the angle subscribed with respect to the positive X-axis) of its neighbor nodes through the beacon messages periodically broadcast by their nodes in the neighborhood.

Data Structures: The data structures used by the generic CDS algorithm is as follows: (i) **CDS-Nodes-List:** This list includes all the nodes that are part of the CDS. (ii) **Covered-Nodes-List:** This list includes all the nodes that are either part of the CDS or is at least a neighbor node of a node in the CDS; (iii) **Priority-Queue:** This list includes all nodes in the Covered-Nodes-List (but not in the CDS-Nodes-List) and are considered the candidate nodes for the next node to be selected for inclusion in the CDS-Nodes-List. To be placed in the priority queue, a node should have at least one uncovered neighbor. Nodes are placed in the Priority-Queue in the decreasing order of node density and node ID for the MaxD-CDS and ID-CDS algorithms respectively and in the increasing order of node velocity for the MinV-CDS algorithm. Any tie between the nodes is broken based on the node IDs (the contending node with a relatively larger ID is placed ahead of the other contending nodes).

Description of Generic CDS Algorithm: The algorithm forms and outputs a list (CDS-Node-List) of all nodes based on a given input graph representing a snapshot of the MANET at a particular time instant. The generic description and pseudo code given here (Figure 1) can be adapted for the particular CDS construction by appropriately incorporating the criteria for choosing the CDS-Select-node for each of the iterations and the criteria for maintaining the Priority-Queue during the different iterations of the algorithm on a particular instance of the network graph.

The node to be included in the CDS-Node-List is the CDS-Select-node at the front of the Priority-Queue, chosen according to the selection criteria of largest node density (MaxD-CDS) or largest node ID (ID-CDS) or minimum node velocity (MinV-CDS), as long as the node has at least one uncovered neighbor. The uncovered neighbors of the CDS node are also added to the Covered-Nodes-List and the Priority-Queue. If the Covered-Nodes-List does not contain all the vertices of the input graph and the Priority-Queue is not empty, we dequeue the Priority-Queue to extract a CDS-Select-node s that is not yet in the CDS-Node-List. All the uncovered neighbor nodes of node s are added to the Covered-Nodes-List and to the Priority-Queue; node s is also added to the

CDS-Node-List. This procedure is repeated until the Covered-Nodes-List contains all the vertices of the graph or the Priority-Queue becomes empty. If the Priority-Queue becomes empty and the Covered-Nodes-List does not have at least one node that is present in the network graph, then the underlying network is considered to be disconnected.

```

Input: Snapshot of the Network Graph  $G = (V, E)$ , where  $V$  is the set of vertices and  $E$  is the set of edges
         CDS-Select-node  $s$  – vertex chosen according to the selection criteria
Auxiliary Variables and Functions:
         CDS-Node-List, Covered-Nodes-List, Priority-Queue, Neighbors(v) for every  $v$  in  $V$ 
Output: CDS-Node-List // contains the list of nodes part of the CDS.
Initialization:
         Covered-Nodes-List =  $\{s\}$ ; CDS-Node-List =  $\Phi$ ; Priority-Queue =  $\{s\}$ 
Begin Construction of CDS

  while ( $|Covered-Nodes-List| < |V|$  and Priority-Queue  $\neq \Phi$ ) do
    CDS-Select-node  $s = Dequeue(Priority-Queue)$ 
    // where  $s \in Covered-Nodes-List$  and  $s \notin CDS-Nodes-List$ 
    // CDS-Select-node  $s$  is selected according to the criteria
    CDS-Node-List = CDS-Node-List  $\cup \{s\}$ 

    for all  $u \in Neighbors(s)$  and  $u \notin Covered-Nodes-List$ 
      Covered-Nodes-List = Covered-Nodes-List  $\cup \{u\}$ 
      Priority-Queue = Priority-Queue  $\cup \{u\}$ 
    end for

    Rearrange the Priority-Queue according to the criteria for ordering the nodes

  end while

  if ( $|Covered-Nodes-List| < |V|$  and Priority-Queue =  $\Phi$ ) then
    return NULL // the network is disconnected and there is no CDS covering all the nodes
  end if

  return CDS-Node-List
End Construction of CDS

```

Figure 1: Pseudo Code for the Generic CDS Construction Algorithm

The time complexity of the generic CDS algorithm can be given as: $\Theta((|E| + |V|) * \log|V|)$ on a network graph of $|V|$ nodes and $|E|$ edges – each of the $|V|$ nodes and their associated edges have to be explored for inclusion in the CDS. If the *Priority-Queue* is implemented as a binary heap, it takes $\Theta(\log|V|)$ time to reorganize the heap after each dequeue and enqueue operation. There could be a total of $|V|$ dequeue operations, one for each of the iterations and a total of $|E|$ enqueue operations, one for every edge that is traversed across all the iterations.

3. SIMULATIONS

The network dimensions are 1000m x 1000m. Each node moves independent of the other nodes under all the three mobility models. The Random Waypoint (RWP) model works by assuming an open network field without any grid constraints. Each node moves randomly from one location to

another location (within the network boundary) with a velocity randomly chosen from $[0..v_{max}]$, where v_{max} is the maximum node velocity. After reaching the targeted location, the node continues to move to another randomly chosen location with a different randomly chosen velocity. The mobility pattern of a node is repeated like this until the end of the simulation.

For the City Section and Manhattan mobility models, the network is divided into grids: square blocks of length (side) 100m, composed of a number of horizontal and vertical streets. Each street has two lanes, one for each direction. A node is allowed to move only along the grids of horizontal and vertical streets. With the City Section model, nodes move from one street intersection to a randomly chosen different street intersection on the shortest path (sequence of street intersections with minimum total distance) with a randomly chosen velocity from $[0..v_{max}]$; after reaching the targeted the street intersection, the node continues to move to another randomly chosen street intersection (on the shortest path) with a randomly chosen velocity. With the Manhattan model, the velocity (chosen from $0..v_{max}$) and direction to move is decided for every street intersection that is come across. If a node can continue to move in the same direction or can also change directions, the node has 50% chance of continuing in the same direction, 25% chance of turning to east/north and 25% chance of turning to the west/south, depending on the direction of the previous movement. If a node has only two options, then the node has an equal (50%) chance of exploring either of the two options. If a node has only one option to move (this occurs when the node reaches any of the four corners of the network), then the node has no other choice except to explore that option. Simulations are conducted for grid block lengths (BL) of 50m, 100m and 200m. The wireless transmission range of a node is 250m.

The network density is varied by performing the simulations with 50 (low-moderate) and 100 (moderate-high) nodes. The v_{max} values used for each of the three mobility models are 5 m/s (about 10 miles per hour) and 50 m/s (about 100 miles per hour), representing levels of low and high node mobility respectively. The pause time is 0 seconds; so, all the nodes are constantly in motion. Each data point in the performance figures shown in this section is an average computed over 5 mobility trace files and 20 randomly selected $s-d$ pairs from each of the mobility trace files. The starting time of each $s-d$ session is uniformly distributed between 1 to 20 seconds.

Overall Simulation Methodology: For each mobility model, snapshots (static graphs) of the network topology are constructed for every 0.25 seconds, starting from time 0 to the simulation time of 1000 seconds. If a CDS is not known at a particular time instant, the appropriate CDS construction algorithm is run on the network snapshot. The CDS determined at a particular time instant is used in the subsequent time instants until the CDS ceases to exist. For a CDS to be considered to exist at a particular time instant, two conditions have to hold good: (i) All the CDS nodes have to stay connected – i.e. reachable from one another directly or through multi-hop paths; and (ii) Every non-CDS node should have at least one CDS node as its neighbor. If a CDS ceases to exist at a particular time instant, the appropriate CDS construction algorithm is again run and the new CDS is continued to be used as explained above. This procedure is continued for the duration of the simulation time.

Performance Metrics: The following two performance metrics are measured under the three mobility models: (i) *Effective CDS Lifetime*: The product of the absolute CDS lifetime (observed from the simulations) and the CDS connectivity. (ii) *CDS Node Size*: This is a time-averaged value for the number of nodes that are part of the CDS used for every time instant over the entire simulation.

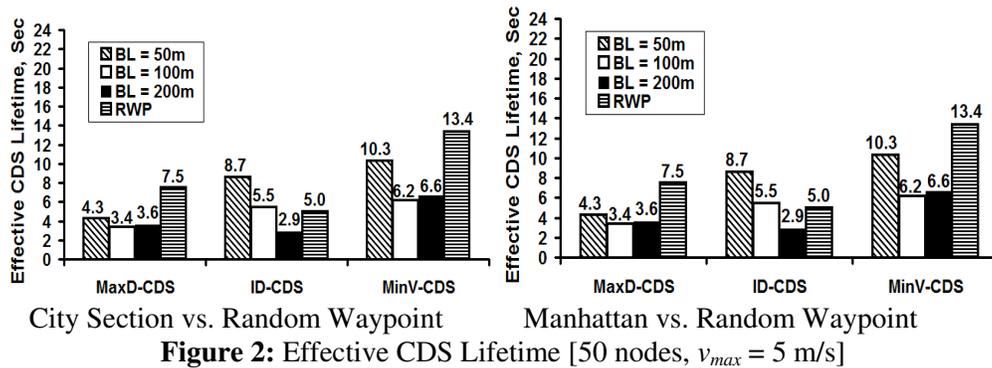


Figure 2: Effective CDS Lifetime [50 nodes, $v_{max} = 5$ m/s]

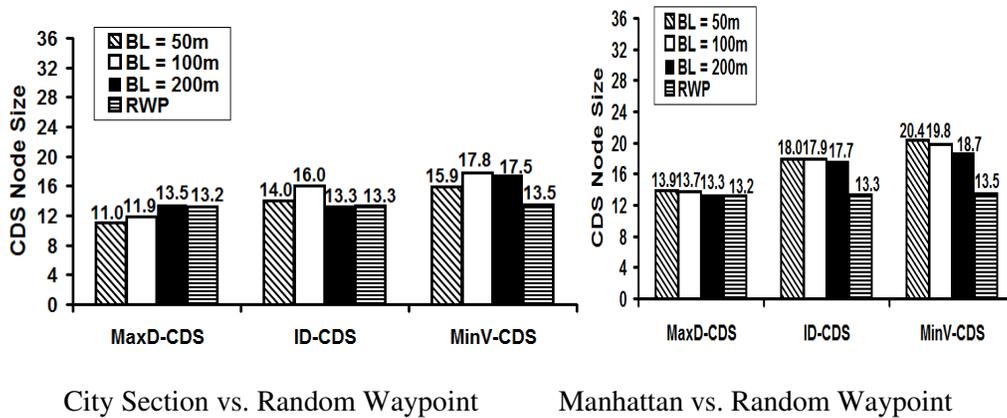


Figure 3: CDS Node Size [50 nodes, $v_{max} = 5$ m/s]

Effective MaxD-CDS Lifetime: The MaxD-CDS incurs the largest lifetime under the RWP model (followed by the Manhattan model) for all the four simulation conditions. For a given simulation condition, it is observed that the difference in the effective MaxD-CDS lifetime incurred with the two VANET mobility models decreases with increase in the block length (from 50m to 200m). It is also observed that for a given block length and node mobility, for most of the simulation conditions, the absolute values for the effective MaxD-CDS lifetime decreases with increase in network density. For both the VANET mobility models, it is observed that the effective MaxD-CDS lifetime decreases with increase in the block length from 50m to 200m. This could be attributed to the nature of the MaxD-CDS algorithm to include only as minimal nodes as possible in the CDS; with a larger block length, even though the number of nodes constituting the MaxD-CDS reduces, the physical distance between the edges that are part of the CDS edges increases. As a result, a MaxD-CDS is more vulnerable to failure when operated in a VANET grid of larger block length.

MaxD-CDS Node Size: The MaxD-CDS incurs the lowest value for the node size under the RWP model for 3 of the 4 simulation conditions (except for the condition of low-moderate density and low node mobility). Thus, it is possible to simultaneously maximize the MaxD-CDS lifetime as well as minimize the node size when operated under the RWP model; there is no significant tradeoff. Between the two VANET mobility models, the MaxD-CDS is observed to sustain a lower node size for the City Section model. This indicates a CDS lifetime-node size

tradeoff between the two VANET mobility models. Under a given simulation condition, the MaxD-CDS incurs a larger effective CDS lifetime under the Manhattan model; but, this is accompanied by a larger CDS node size. The relative difference in the MaxD-CDS node size between the Manhattan and City Section models could be as large as 25%, and the relative difference decreases with increase in the grid block length from 50m to 200m. For the RWP model, as the network density is doubled from 50 to 100 nodes for a given mobility level, the MaxD-CDS node size decreases by about 10-25%. This decrease, rather than an increase that is typically expected, could be attributed to the nature of the MaxD-CDS algorithm to include only those nodes as part of the CDS so that as many non-CDS nodes could be covered as possible. The negative impact of this greedy approach is observed in the effective CDS lifetime that decreases with increase in network density, due to the need to cover an increased number of non-CDS nodes with a reduced number of CDS nodes at high network density. The MaxD-CDS node size is not significantly affected with increase in network density under the two VANET mobility models.

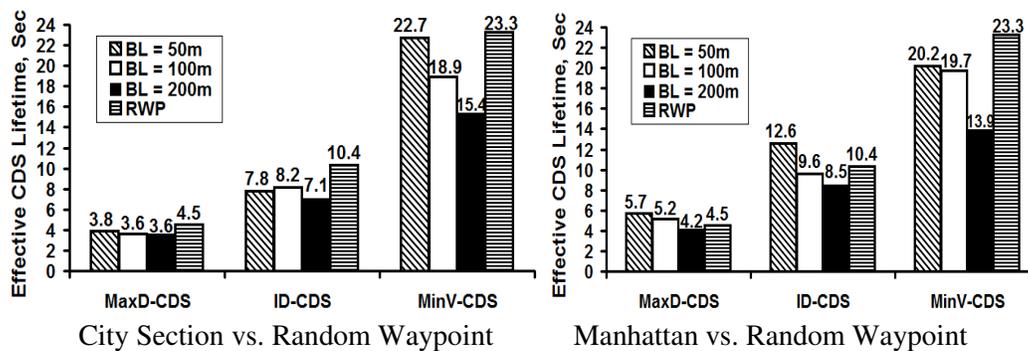


Figure 4: Effective CDS Lifetime [100 nodes, $v_{max} = 5$ m/s]

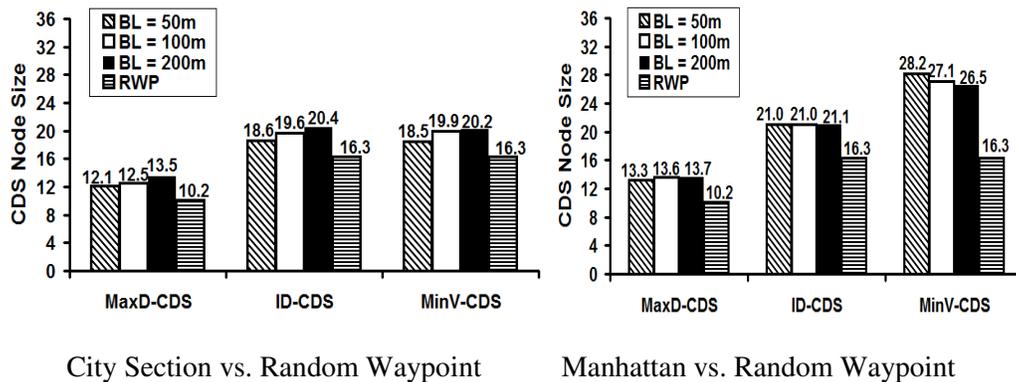


Figure 5: CDS Node Size [100 nodes, $v_{max} = 5$ m/s]

Effective ID-CDS Lifetime: The ID-CDS incurs a larger effective lifetime under the two VANET mobility models compared to the RWP model under all the simulation conditions. The effective ID-CDS lifetime was the largest under the Manhattan model (with a block length of 50m) for 3 of the 4 operating conditions and under the City Section model (block length of 50 and 200m) in scenarios with low-moderate network density and moderate-high node mobility. Overall, it is also observed that for the two VANET models, the effective ID-CDS lifetime decreases with increase in the grid block length from 50m to 200m.

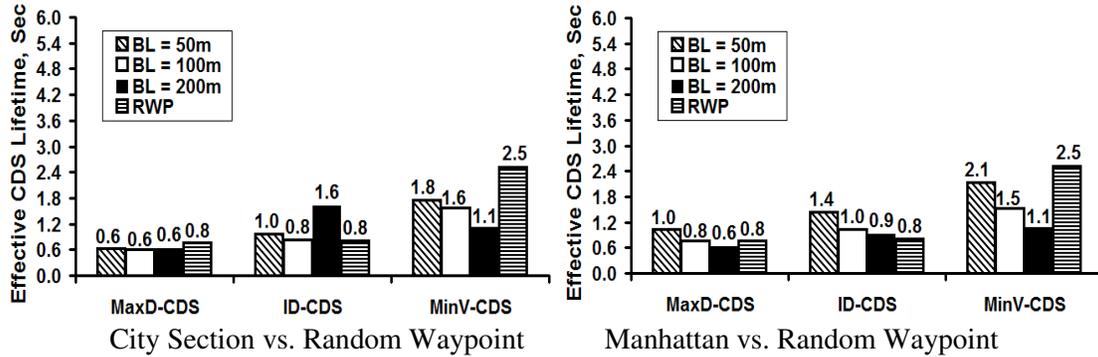


Figure 6: Effective CDS Lifetime [50 nodes, $v_{max} = 50$ m/s]

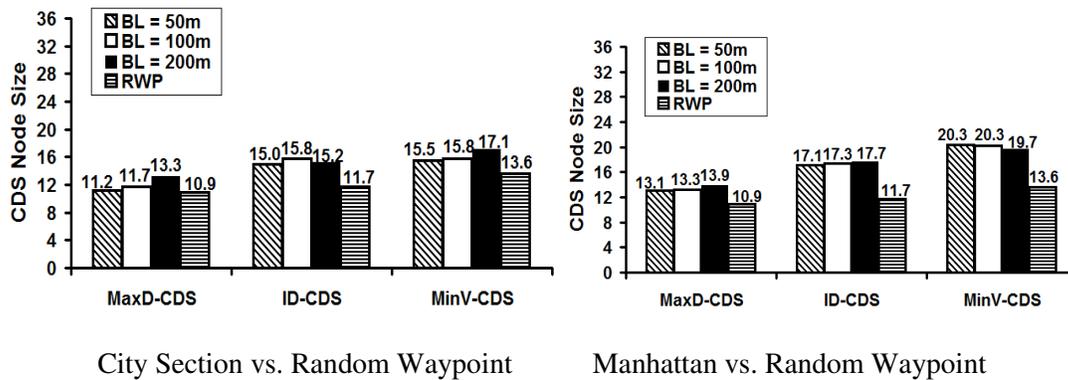
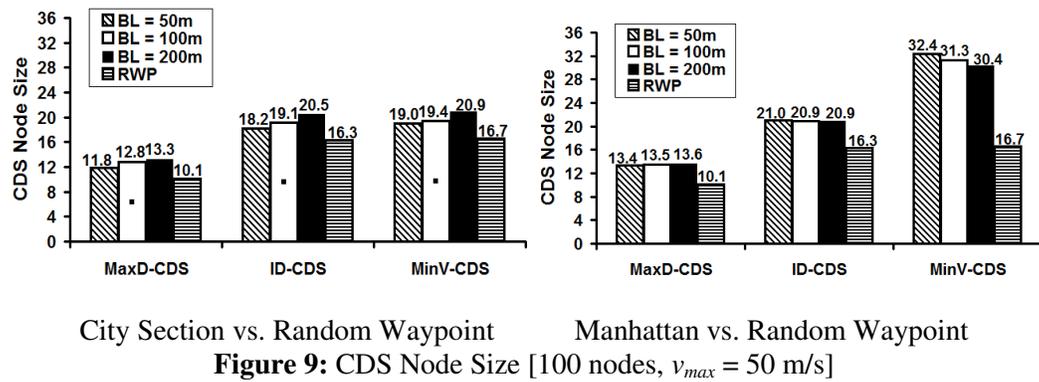
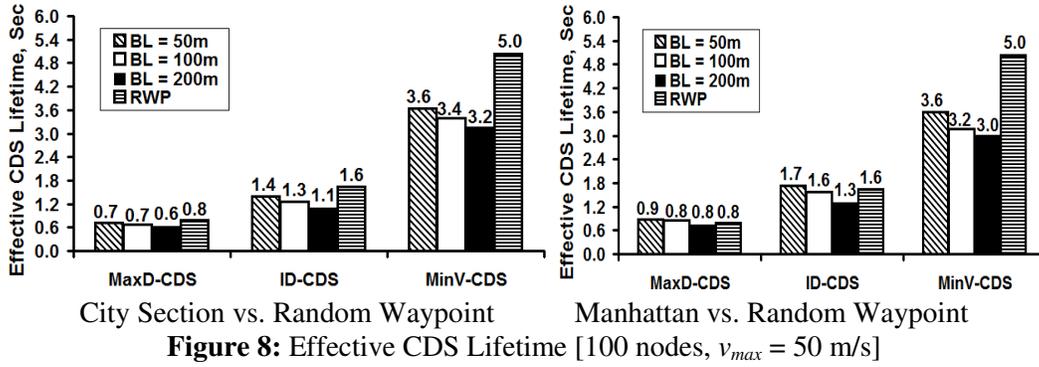


Figure 7: CDS Node Size [50 nodes, $v_{max} = 50$ m/s]

ID-CDS Node Size: The ID-CDS exhibits a significant lifetime-node size tradeoff. The ID-CDS node size is the lowest under the RWP model and is the largest under the Manhattan model that also yielded the largest effective ID-CDS lifetime. Thus, it is not possible to simultaneously maximize the effective ID-CDS lifetime by incurring a lower node size under any mobility model. The ratio of the maximum ID-CDS node size to that of the minimum node size ranges from 1.2 to 1.8; the difference increases with increase in the network density and/or node mobility. Between the two VANET mobility models, the ID-CDS node size is larger when run the Manhattan mobility model compared to the City Section model. Another interesting observation is that the grid block length does not significantly impact the ID-CDS node size under both the VANET mobility models.

Effective MinV-CDS Lifetime: The MinV-CDS incurs the largest effective lifetime under the RWP model for all the simulation conditions. The difference in the effective MinV-CDS lifetime between the RWP model and that of the two VANET mobility models decreases with increase in the network density (for a fixed node mobility) and increases with increase in node mobility (for a fixed network density). Between the two VANET mobility models, it is observed that the effective MinV-CDS lifetime incurred under the Manhattan model is greater (by about 20%) than that incurred under the City Section model in low-moderate network density and low-moderate node mobility scenarios. With increase in the network density and/or node mobility, the effective MinV-CDS lifetime incurred under the City Section model is larger (at most by about 25%) compared to that incurred under the Manhattan model. The grid block length significantly influences the effective MinV-CDS lifetime under both the VANET mobility models. The

effective MinV-CDS lifetime incurred under a grid block length of 50m could be as large as twice the grid block lengths of 100m and 200m. This could be attributed to the presence of physically longer edges in the MinV-CDS determined under larger values of the grid block lengths.



MinV-CDS Node Size: The MinV-CDS node size is the lowest under the RWP model for all the simulation conditions. Thus, one could obtain a larger MinV-CDS lifetime and a lower MinV-CDS node size when operated under the RWP model. The difference (ranging from 25% to 100%) in the MinV-CDS node size incurred with the RWP model and the two VANET mobility models increases predominantly with increase in the network density and with increase in the grid block length. Between the two VANET mobility models, it is observed that the MinV-CDS node size is larger when run under the Manhattan mobility model for all the simulation conditions, with the difference increasing with increase in network density and/or node mobility. For a given condition of network density and node mobility, the difference in the MinV-CDS node size decreases with increase in the grid block length from 50m to 200m. Between the two VANET mobility models, it is possible to simultaneously optimize both the MinV-CDS lifetime and the node size when operated under the City Section model vis-à-vis the Manhattan model.

4. CONCLUSIONS

The MinV-CDS incurs the largest effective lifetime under all the three mobility models; and at the same time incurs the largest node size, indicating a CDS lifetime-node size tradeoff. The MaxD-CDS and MinV-CDS incur the largest effective lifetime and lowest node size under the RWP model vis-à-vis the two VANET mobility models. On the other hand, the ID-CDS incurs the largest effective lifetime under the VANET mobility models (predominantly with the

Manhattan model) and the lowest node size with the RWP model. Thus, there is a CDS lifetime-node size tradeoff with respect to using an appropriate mobility model for the ID-CDS. With regards to the two VANET mobility models, the MaxD-CDS and ID-CDS yielded the largest effective lifetime under the Manhattan model for most of the simulation conditions; whereas the MinV-CDS yielded the largest effective lifetime under the City Section model for a majority of the simulation conditions. On the other hand, all the three connected dominating sets sustained a larger CDS node size under the Manhattan mobility model compared to the City Section model. This illustrates a CDS lifetime – node size tradeoff for the MaxD-CDS and ID-CDS with respect to the choice of usage between the two VANET mobility models. With regards to the impact of the grid block length on the stability of the connected dominating sets, one can conclude that the lower grid block lengths yielded more stable connected dominating sets compared to larger grid block lengths. Also, there is no appreciable increase in the CDS node size when operated at lower grid block lengths for all the three CDS algorithms under both the VANET mobility models.

REFERENCES

- [1] Broch, J., Maltz, D. A., Johnson, D. B., Hu, Y. C., and Jetcheva, J.: A Performance Comparison of Multi-hop Wireless Ad hoc Network Routing Protocols. In: International Conference on Mobile Computing and Networking, pp. 85--97, ACM, Dallas, USA (1998)
- [2] Siva Ram Murthy, C., and Manoj, B. S.: Ad hoc Wireless Networks – Architectures and Protocols, Prentice Hall, USA (2004).
- [3] Sinha, P., Sivakumar, R., and Bhargavan. V.: Enhancing Ad hoc Routing with Dynamic Virtual Infrastructures. In: The 20th International Conference on Computer and Communications Societies, pp. 1763--1772, IEEE, Anchorage, Alaska, USA (2001).
- [4] Wang, F., Min, M., Li, Y., and Du, D.: On the Construction of Stable Virtual Backbones in Mobile Ad hoc Networks. In: The International Performance Computing and Communications Conference (IPCCC), IEEE, Phoenix, Arizona, USA (2005).
- [5] Meghanathan, N.: An Algorithm to Determine the Sequence of Stable Connected Dominating Sets in Mobile Ad Hoc Networks. In: The 2nd Advanced International Conference on Telecommunications, IARIA, French Caribbean (2006).
- [6] Velumylum, N., and Meghanathan, N.: On the Utilization of ID-based Connected Dominating Sets for Mobile Ad hoc Networks, International Journal of Advanced Research in Computer Science, 1, 3, 36--43 (2010).
- [7] Meghanathan, N.: An Algorithm to Determine Minimum Velocity-based Stable Connected Dominating Sets for Ad hoc Networks, Communications in Computer and Information Science Series, 94, 206--217 (2010).
- [8] Bettstetter, C., Hartenstein, H., and Perez-Costa, X.: Stochastic Properties of the Random-Way Point Mobility Model, Wireless Networks, 10, 5, 555--567 (2004).
- [9] Camp, T., Boleng, J., and Davies, V.: A Survey of Mobility Models for Ad hoc Network Research, Wireless Communication and Mobile Computing, 2, 5, 483--502 (2002).
- [10] Kuhn, F., Moscibroda, T., and Wattenhofer, R.: Unit Disk Graph Approximation. In: The Workshop on Foundations of Mobile Computing, Philadelphia, USA (2004).
- [11] Hofmann-Wellenhof, B., Lichtenegger, H., and Collins, J.: Global Positioning System: Theory and Practice, Springer Wien, New York, USA (2004).