Meghanathan, Natarajan; Thompson, Joseph
On the Different Forms of Spanning Tree-based Broadcast Topologies for Mobile Ad hoc Networks
International Journal of Combinatorial Optimization Problems and Informatics, vol. 4, núm. 1, enero-abril, 2013, pp. 3-11
International Journal of Combinatorial Optimization Problems and Informatics
Morelos, México

Available in: http://www.redalyc.org/articulo.oa?id=265225625002
On the Different Forms of Spanning Tree-based Broadcast Topologies for Mobile Ad hoc Networks

Natarajan Meghanathan¹ and Joseph Thompson²
¹Jackson State University, USA ²Mississippi State University, USA
Corresponding Author E-mail: natarajan.meghanathan@jsms.edu

Abstract. In this paper, we study the different forms of spanning tree-based topologies that could be used for efficient broadcasting in mobile ad hoc networks (MANETs). In this context, we consider the minimum distance-based, predicted link expiration time-based and the minimum velocity-based broadcast topologies, respectively referred to as MD-BT, LET-BT and MV-BT. The LET and minimum velocity-based broadcast topologies are characteristic of MANETs as the two topologies are constructed using link criterion that are based on the mobility of the nodes. We conduct extensive simulation study of these three broadcast topologies for different conditions of network density and node mobility. We identify the topology that could be used to optimize MANET performance with respect to critical quality-of-service parameters such as energy efficiency, stability and delay, measured respectively through metrics such as the edge distance ratio, tree lifetime and diameter. We observe tradeoffs between the different broadcast topologies with respect to the above three performance metrics. The MD-BT, LET-BT and MV-BT are best to optimize energy efficiency, stability and diameter respectively. There is no single broadcast topology that could optimize all the above three performance metrics for MANETs.

Keywords: Broadcast Topology, Stability, Energy Efficiency, Diameter, Simulations, Mobile Ad hoc Networks.

1 Introduction

A mobile ad hoc network (MANET) is a dynamic distributed system of arbitrarily moving wireless nodes operating under limited battery charge and bandwidth. Each node operates with a limited transmission range and be able to directly communicate only with nodes within its transmission range (referred to as neighbors). As such, a node does not know the entire topology and often indulges in a broadcast-based query-reply mechanism (commonly referred to as flooding) to discover routes to distant nodes (nodes that are outside its neighborhood) in order to send data packets. Broadcast is a communication paradigm wherein a message originating from one node reaches every other node in the network [1]. However, due to the distributed nature of MANETs, broadcast is often realized through the active involvement of all the nodes in the network in terms of forwarding the message in each other’s neighborhood.

If all the links in a MANET are used for broadcasting a message, it would result in too much of redundancy (a node would receive a copy of the same message from each of its neighbors) as well as energy consumption and collision of messages. This is referred to as the Broadcast Storm problem [2]. Several optimizations (such as probability-based, counter-based, area-based, distance-based, cluster-based and multi-point relay based mechanisms) [2] for broadcasting in MANETs to avert the broadcast storm problem have been proposed in the literature. A common thread among all of these broadcast optimization mechanism is to choose only a subset of the nodes in the network to broadcast a message (in its neighborhood) so that the message reaches every node in the network. The broadcast optimization mechanisms only differ in the criteria used to select the set of nodes that forward a message to their neighbors. However, we find a fundamental flaw in such broadcast optimization mechanisms. The nodes that broadcast a message (to their neighbors) end up losing more energy than nodes that do not broadcast. In addition, given that broadcast-based flooding is the common mechanism used to discover routes in MANETs [3], the nodes that broadcast a message end up serving as intermediate nodes for unicast and multicast sessions that could involve the transfer of several data packets. This could again result in the overuse of certain nodes at the cost of others [4][5].
In this paper, we envision an alternate broadcast optimization mechanism that involves only the use of certain links (instead of certain nodes as in the currently used optimization techniques) to ensure fairness of node usage and actively involve every node in the network in the broadcast process. We leverage the concept of ‘Spanning Trees’ from graph theory for this purpose. A spanning tree on a graph of \( n \) vertices is a sub graph that includes all the \( n \) vertices in the original graph that are connected through only \( n - 1 \) edges [6]. We model a MANET as a unit disk graph [7] wherein each node is a vertex and an edge exists between two vertices if and only if the physical distance (referred to as the Euclidean distance in planar graphs) between the two corresponding nodes is less than or equal to the transmission range of the nodes. Since a spanning tree connects all the nodes in the network, a message originating at one node and broadcast through the edges of a spanning tree (i.e., forwarded by all the nodes along the edges of the spanning tree) will reach every other node in the network. We refer to a broadcast topology as the network of links forming the edges of the spanning tree. A spanning-tree based broadcast topology will ensure active participation from all nodes in the network in the broadcast process; and at the same time, each node is guaranteed to receive the message exactly once. Note that, we completely avoid redundant transmissions and receptions, as a spanning tree does not involve any cycle or multiple paths between any two nodes in the network.

From a graph theory perspective, spanning trees are typically studied as minimum-weight or maximum-weight spanning trees determined on weighted graphs. The weight of a spanning tree is the sum of its edge weights. In this paper, we will apply the notion of edge weights to the MANET links and determine appropriate minimum and maximum weight spanning trees that can be used for efficient broadcasting. We propose the use of three different spanning-tree broadcast topologies for MANETs: (i) A minimum-distance based broadcast topology (MD-BT) wherein the edge weights of the unit disk graph refer to the physical Euclidean distance between the constituent nodes of an edge, and a MD-BT is a minimum-weight spanning tree; (ii) A link expiration time (LET)-based broadcast topology (LET-BT) wherein the edge weights of the unit disk graph correspond to the predicted expiration time of the links (determined according to the well-known model proposed in [8]) corresponding to the edges of the unit disk graph, and a LET-BT is a maximum-weight spanning tree and (iii) A minimum velocity-based broadcast topology (MV-BT) wherein the weight of an edge in the unit disk graph is the maximum of the velocity of the two nodes constituting the corresponding link in the MANET, and the MV-BT is a minimum-weight spanning tree.

We study the three spanning tree-based broadcast topologies under diverse conditions of network density and node mobility. Our simulations are conducted on unit disk graphs captured as network snapshots at regular time instants (for every 0.25 seconds) over a simulation period (1000 seconds). To start with, we determine a broadcast topology (according to a particular spanning tree algorithm) on the initial network topology and validate its existence in the subsequent network snapshots. A spanning tree-based broadcast topology exists at a particular time instant if all the edges of the tree exist in the underlying network graph corresponding to the snapshot captured at that time instant. If the currently known spanning tree no longer exists, we repeat the above process (i.e., determine a new spanning tree and use it as long as it exists). We apply the same procedure for each of the three spanning tree-based broadcast topologies. We compare them with respect to three critical quality-of-service parameters for MANETs: energy-efficiency, stability and delay, evaluated respectively using performance metrics such as edge distance ratio (average of the ratios of the Euclidean distance to the transmission range of the edges), lifetime of the spanning tree (a spanning tree ceases to exist even if one of its edges break) and diameter (the maximum number of edges between any two nodes in a spanning tree).

The rest of the paper is organized as follows: Section 2 gives more information about the three broadcast topologies evaluated in this research and the spanning tree algorithms behind them. Section 3 presents the simulation results evaluating the three spanning-tree based broadcast topologies and interprets them. Section 4 concludes the paper. For simplicity, we assume a homogeneous network wherein all nodes operate under an identical transmission range. However, the analysis conducted in this research can be applied to heterogeneous networks too. For the rest of the paper, we use the terms ‘node’ and ‘vertex’ as well as ‘link’ and ‘edge’ interchangeably. They mean the same.

2 Related Work

For all the three broadcast topologies explored in this research, the MANET is modeled as a unit disk graph with the nodes representing the vertices and the links between these nodes captured as edges. There exists an edge between two vertices in a graph if and only if the physical Euclidean distance between the constituent end nodes of the link is within the transmission range of the nodes. The weights assigned to the edges differ for each of the three broadcast topologies (as described below) and an appropriate spanning tree algorithm is thereby used to determine the topology.
2.1 Spanning Tree-based Broadcast Topologies

For the MD-BT spanning trees, the weight of an edge is the physical Euclidean distance between the two end nodes of the corresponding link in the MANET. For energy efficiency, links of shorter physical distance are preferred over links of larger physical distance. Hence, on a weighted unit disk graph modeling the edge weights as link distances, we run the minimum spanning tree algorithm (Kruskal’s algorithm) [6] to determine a minimum-weight spanning tree that spans all the vertices of the graph with the lowest value for the sum of the link distances. Figure 1 presents the pseudo code of the Kruskal’s algorithm used to compute the minimum-weight spanning tree in this research. Run-time complexity of Kruskal’s algorithm is O(n^2 log n) on a network graph of n nodes [6].

---

Begin Algorithm Kruskal (G = (V, E))
  A Ì Ø // Initialize the set of edges to null set
  for each vertex v_i ∈ V do
    Component (v_i) Ì i
  Sort the edges of E in the non-decreasing (increasing) order of weights
  for each edge (v_i, v_j) ∈ E, in order by non-decreasing weight do
    if (Component (v_i) ≠ Component (v_j) ) then
      A Ì A U (v_i, v_j)
      if Component(v_i) < Component(v_j) then
        for each vertex v_k in the same component as of v_j do
          Component(v_k) Ì Component(v_i)
      else
        for each vertex v_k in the same component as of v_i do
          Component(v_k) Ì Component(v_j)
    end if
  end for
  return A
End Algorithm Kruskal

Fig. 1. Pseudo Code for the Kruskal’s Minimum-weight Spanning Tree Algorithm

Figure 2: Example: Execution of the Minimum-Weight Spanning Tree Kruskal’s Algorithm
Figure 2 illustrates the construction of minimum-weight spanning tree using Kruskal’s algorithm. The label outside a circle indicates the node ID and the label inside a circle indicates the Component (Component ID) to which the node belongs to. Initially, to start with, each node is in its own component and with the inclusion of an edge into the spanning tree, two components merge and all the nodes within the merged component get a component ID which is the lower of the component IDs of the two components before the merger. For example, if all the nodes in a component had a component ID A and all the other nodes in another component had a component ID B, then when an edge whose end nodes are in these two components is included into the spanning tree, then the two components are merged and all the nodes in the merged component get component ID of A, which is the alphabetically (lexicographically) lower of the two component IDs A and B. The edges that get selected in the successive iterations for inclusion in the spanning trees are thickened and kept track of throughout the execution of the algorithm.

2.2 Link Expiration Time-based Broadcast Topology (LET-BT)

For the LET-BT spanning trees, the weight of an edge is the predicted expiration time of the corresponding link in the MANET. The predicted link expiration time (LET) of a link \( i - j \) between two nodes \( i \) and \( j \), currently at \((X_i, Y_i)\) and \((X_j, Y_j)\), and moving with velocities \( v_i \) and \( v_j \) in directions \( \theta_i \) and \( \theta_j \) (with respect to the positive X-axis) is computed using the formula (equation 1) [8]:

\[
\text{LET}(i, j) = \frac{-(ab + cd) + \sqrt{(a^2 + c^2)R^2 - (ad - bc)^2}}{a^2 + c^2}
\]

where \( a = v_i^* \cos \theta_i - v_j^* \cos \theta_j; b = X_j - X_i; c = v_i^* \sin \theta_i - v_j^* \sin \theta_j; d = Y_j - Y_i \)

A link with a larger LET is likely to be more stable (i.e. exist for a longer time) compared to a link with a lower LET. Hence, we hypothesize that a spanning tree with a larger value of the sum of the LETs of its constituent edges is more likely to exist for a longer time than a spanning tree with a smaller value for the sum of the LETs of its edges. Under this hypothesis, we determine a maximum-weight spanning tree of the unit disk graph with the LETs as the edge weights. To do so, we utilize the Kruskal’s minimum-weight spanning tree algorithm shown in Figure 1. The only change we have to do to compute a maximum-weight spanning tree is to change the sign of all the LET-edge weights (in the unit disk graph) to negative, and then call the Kruskal’s minimum-weight spanning tree algorithm; at the end, we again revert the sign of the edges in the minimum-weight spanning tree to positive and the resulting tree of positive LET-edge weights is the desired maximum-weight spanning tree for the LET-BT.

2.3 Minimum Velocity-based Broadcast Topology (MV-BT)

For the MV-BT spanning trees, the weight of an edge is the maximum of the velocities of the two constituent end nodes of the corresponding link in the MANET. A link with slow moving nodes is more likely to exist for a longer time compared to a link with fast moving nodes. Under this conjecture, we hypothesize that a spanning tree with a lower value for the sum of its edge weights (that model that velocities of the constituent end nodes of the links) is likely to be more stable compared to a spanning tree with a larger value for the sum of its edge weights. The spanning tree for the MV-BT can be constructed by running the Kruskal’s algorithm on a unit disk graph with the edge weights set as the maximum of the velocities of the two constituent end nodes of the edge.

3 Simulations

We implemented the algorithms to determine the MD-BT, LET-BT and MV-BT spanning trees in a discrete-event simulator developed by us in Java. This simulator has been successfully used in several of our previous research studies [9][10] related to centralized implementation of graph theory algorithms to simulate communication protocols for MANETs.

3.1 Network Density

The dimensions of the network topology are 1000 meters x 1000 meters. The network density is represented as a measure of the average neighborhood size, which is calculated as follows: \( N \times R^2 / A \), where \( N \) is the number of nodes in the network, \( R \) is the transmission range of a node, and \( A \) is the network area. The transmission range per node in all of our simulations is 250 meters. Using a fixed transmission range and network area, we vary network density between levels of low, medium, and high,
by altering the number of nodes (N) in the network. In our simulations, we use networks comprising 25 and 50 nodes (low density), 100 nodes (moderate density), and 150 nodes (high density).

3.2 Node Mobility Model

We use the well-known Random Waypoint model [11] as the mobility model for our simulations according to which the nodes move independent of each other. To start with, we randomly distribute the nodes inside the 1000m x 1000m network. Each node chooses a destination location to move within the network at a speed randomly chosen from the range \([0, \ldots, v_{\text{max}}]\). After reaching the targeted destination location, the node continues to move by choosing another location and moves to that location with a random speed newly chosen from the range \([0,\ldots, v_{\text{max}}]\). A waypoint refers to the movement of a node with a fixed speed (randomly chosen from the above range) in a particular direction from a starting location to a chosen destination location. The location and speed chosen by a node for each waypoint is independent of the mobility history (i.e., the previous waypoints). The mobility profile of a node refers to the sequence of waypoints and the associated location, velocity and time information.

We simulate the mobility of the nodes offline and generate the mobility profile of the nodes for a particular condition of network density (i.e. number of nodes) and node mobility (i.e. \(v_{\text{max}}\) value). The values of \(v_{\text{max}}\) are varied between 5 m/s, 25 m/s, and 50 m/s to represent levels of low, medium and high mobility respectively. Under each of the above simulation conditions, we generate mobility profile files for a simulation time of 1000 seconds.

3.3 Simulation Methodology

Our simulation methodology is as follows: The mobility profile of the nodes, generated offline as described above, is passed as input to our simulation code for the spanning tree algorithms. We collect snapshots of the network topology for every 0.25 seconds, spanning over a simulation time period of 1000 seconds. The simulations are conducted independently for each of the three spanning tree algorithms. Note that a spanning tree of the nodes exists only if the underlying unit disk network graph is connected. For a particular spanning tree algorithm (say MD-BT), to start with, we determine the MD-BT spanning tree on the initial network snapshot (at time instant 0 sec). If we could find a MD-BT spanning tree, we continue to use it during the subsequent time instants (0.25 sec, 0.5 sec, 0.75 sec, etc) as long as it exists in the network snapshots for those time instants. To validate the existence of a spanning tree in a particular network snapshot, we have to determine the locations of the nodes (using their mobility profile) at the particular time instant and then determine if the physical Euclidean distances between the constituent nodes of the different edges in the spanning tree are within the transmission range. A spanning tree is considered broken even if one of the edges in the tree ceases to exist (i.e., the physical Euclidean distance of the edge is greater than the transmission range). In such a case, we determine a new MD-BT spanning tree on the network snapshot for the time instant at question. We continue to use the new tree as long as it exists, and the above procedure is repeated for the entire duration of the simulation. The same procedure is applicable to determine the sequence of LET-BT and MV-BT spanning trees.

3.4 Performance Metrics

We evaluated the performance of the MD-BT, LET-BT and MV-BT spanning tree algorithms with respect to three critical quality-of-service parameters such as the energy efficiency, stability and delay.

Edge Distance Ratio: We evaluate the energy efficiency of the three spanning tree algorithms by computing a percentage distribution of the edge distance ratio of the different edges in the spanning tree. For MANET simulations, the energy lost to transmit data is usually modeled as the fourth power of the distance [12]. In this context, we introduce a dimensionless metric called the edge distance ratio, defined as the ratio of the physical Euclidean distance corresponding to an edge and the maximum transmission range of the two constituent nodes corresponding to the edge. Since in this research, we assume a homogeneous network of nodes, the transmission range is the same for all the nodes in the network. As a result, the edge distance ratio is simply the ratio of the Euclidean distance between the two end nodes of the edge to that of the transmission range per node. Larger the edge distance ratio of an edge, the larger is the transmission energy loss for the edge. For a particular condition of network density and node mobility, the edge distance ratios shown (in Figures 3, 4 and 5) for any spanning tree is the average of the edge distance ratio values measured for the tree at the time instants during which we determined a new spanning tree (i.e., transitioned from one spanning tree to another) across the entire simulation time period.

Spanning Tree Lifetime: The spanning tree lifetime is a measure of the stability of the broadcast topology. We measure the average of the lifetimes of the individual spanning trees determined across the duration of the simulation. The larger is the lifetime, the more stable is the broadcast topology and vice-versa.
Spanning Tree Diameter: The diameter of a spanning tree is the time-averaged value of the maximum number of hops between any two nodes in the individual trees (according to a particular algorithm) determined and used over the entire simulation time period. For example, if we had used a sequence of three spanning trees for a total simulation duration of 20 seconds, existing for time instants 0 – 10, 10 – 13, 13 – 20 seconds and are of diameters 7, 9 and 8 respectively, then the time-averaged diameter for the entire simulation time period is \((10\times7 + 3\times9 + 7\times8)/ 20 = 7.65\) and not simply 8.0 (the average the diameters 7, 9 and 8). A broadcast topology with a smaller diameter is preferred for paths with lower hop count and end-to-end delay.

3.5 Percentage Distribution of the Edge Distance Ratio

The edge distance ratio is a direct measure of the energy efficiency of a spanning tree for network-wide broadcasting, and also has an impact on the lifetime of the spanning trees, as discussed in Section 3.6. In the performance figures 3, 4 and 5, we analyze the percentage distribution of the distance ratio of the edges in the three spanning trees for different network density values. To reduce the computation overhead involved, we use only the mobility profile generated for medium mobility condition \((v_{\text{max}} = 25 \text{ m/s})\) to determine the edge distance ratios.

As expected, the MD-BT has the lowest tree edge distance ratio – attributable to the distance criterion integral to the selection of the edges to be part of the broadcast topology. From Figure 3, we notice that as the network density increases from 25 nodes to 150 nodes, the percentage of edges in the MD-BT with distance ratio less than 0.4 increases from 38% to 86%, virtually knocking out any edge with a distance ratio of more than 0.8 (in a MD-BT spanning 150 nodes).

Among the other two broadcast topologies (MV-BT and LET-BT), the LET-BT had a lower tree edge distance ratio. This could be justified with the selection of edges having a relatively larger predicted link expiration time (LET) to be part of the broadcast topology. Such edges cannot have too large of a physical Euclidean distance close to the transmission range per node; the distance ratio of the edges selected to be part of the LET-BT is normally within the 0.5 to 0.75 range. Thus (even if not directly embedded into the edge selection criterion), by preferring to include edges with a larger LET, the distance criterion is somewhat embedded into the choice of the edges selected for the broadcast topology. However, unlike the MD-BT, the percentage distribution of the distance ratios of the edges constituting a LET-BT does not change significantly with network density.
The MV-BT incurs the largest values for the edge distance ratio among the three broadcast topologies and this is attributable to the “zero” consideration given to the Euclidean distance between the constituent edges of the tree. MV-BT is 100% velocity-based; even if two nodes are moving parallel to each other (and are likely to remain as neighbors for some more time in the future) or even if two nodes are approaching each other, if the velocity of the at least one of the two end nodes of the edge is larger, the weight of the edge ends up being a large value. Edges with lower velocities for the end nodes need not have a lower distance ratio and this is evident in the Edge Distance Ratio values computed for the MV-BT. As it appears, the distance ratio of about 1/3rd of the edges that are chosen to be part of the MV-BT are more than 0.8 (at the time of constructing the topology) and this has a significant impact on the lifetime of the MV-BT. Like LET-BT, the MV-BT is also insensitive to the changes in the network density vis-à-vis the distance ratio of the edges constituting the spanning tree. On the other hand, even in low density networks, less than 10% of the edges chosen to be part of the MD-BT have a distance ratio greater than 0.8; in high-density networks (150 nodes), there are virtually no edges of the MD-BT that have a distance ratio greater than 0.8 at the time of construction of the topology.

3.6 Spanning Tree Lifetime

We notice the LET-BT to have the longest lifetime among the three broadcast topologies and the MV-BT to have the lowest lifetime. The performance of the LET-BT vis-à-vis the tree lifetime could be attributed to the criteria according to which the edges are chosen (the edges having a larger value for the predicted LET are preferred for inclusion into the spanning tree). The difference in the lifetimes of the LET-BT and MD-BT is significantly high at low network densities and it gradually reduces and the two spanning tree topologies have almost the same lifetime in high-density networks (150 nodes). This could be attributed to the increase in the number of more stable edges (edges with distance ratio < 0.4) in the MD-BT with increase in network density. In an earlier work [13], we had proposed a minimum velocity-based connected dominating set (CDS) algorithm to determine a sequence of stable CDS for MANETs. In this research, we envisioned a similar approach of using the node velocities to determine a stable spanning tree for MANETs. However, the idea of using node velocity as the edge weight to construct the spanning tree did not work as expected. The MV-BT trees have the lowest lifetime and they decrease with increase in network density. Except for a proportional drop in the magnitude, for a given network density, we do not see any variation in the performance of the three broadcast topologies with respect to changes in node mobility (refer Figure 6).
3.7 Spanning Tree Diameter

We observe the MD-BT to have the largest diameter (refer Figure 7) and the MV-BT to have the smallest diameter. Since, the MD-BT prefers edges with a lower distance ratio and it is a spanning tree (there can be only $n-1$ edges for an $n$-node network), there is likely to be relatively more intermediate nodes on paths between any two nodes on the minimum-distance based spanning tree broadcast topology. On the other hand, as observed in Figure 5, the MV-BT comprises of edges with about $1/3^{rd}$ of them having a distance ratio greater than 0.8. Hence, the paths between any two nodes on the minimum-velocity based spanning tree broadcast topology will involve relatively fewer intermediate nodes. The diameter of LET-BT spanning tree is at most 40% more than that of a MV-BT.

For a given network density, we do not see any appreciable impact of node mobility on the diameter for the three spanning tree broadcast topologies. On the other hand, for a given condition of node mobility, the diameter of the individual broadcast topologies increases with increase in network density. This is vindicated by the fact that a spanning tree has to cover all the nodes in the network. However, an interesting observation is that difference in the diameters of the three spanning trees increases with increase in network density. For example, in low-density networks, diameter of a LET-BT is about 6-15% more than the diameter of a MV-BT; whereas, in moderate and high density networks, the diameter of a LET-BT is about 30% and 40% more than the diameter of the MV-BT. Similar observations can be made on the differences in the diameter of the MD-BT and the MV-BT with increase in the network density.

Fig. 7.1. $v_{\text{max}} = 5$ m/s  
Fig. 7.2. $v_{\text{max}} = 25$ m/s  
Fig. 7.3. $v_{\text{max}} = 50$ m/s

4 Conclusions and Future Work

The high-level contribution of this paper is a simulation-based analysis on the effectiveness of the spanning tree-based topologies for efficient broadcasting in MANETs. In addition to the traditionally used minimum distance-based spanning trees, we have proposed two new forms of spanning trees that are characteristic of MANETs. These are the predicted link expiration time (LET)-based and minimum velocity (MV)-based spanning trees. While the LET-based spanning tree could be determined as a maximum weight spanning tree on a unit disk graph modeling the edge weights as the predicted LETs, the MV-based spanning tree could be determined as a minimum-weight spanning tree on a unit disk graph with the weight of an edge modeled as the maximum of the velocity of the two end nodes constituting the edge. We evaluated the energy efficiency (in terms of the edge distance ratio), stability (in terms of the lifetime) and diameter (a measure of the delay) of the broadcast topologies based on these three spanning trees.

We observe a tradeoff among these three broadcast topologies with respect to the above performance metrics. We could not identify one spanning tree that optimizes all of the above three performance metrics. The MD-BT incurs the lowest values for the edge distance ratio, indicating that the broadcast topologies based on the minimum distance-based spanning trees could be the most energy efficient. The LET-BT spanning trees were the most stable; however, these trees had a larger value for the edge distance ratio as well as had a larger diameter. The MV-BT trees also had an edge distance ratio (similar to that of the LET-BT); but they were not stable (contrary to our expectation that the incorporation of the node velocities in the edge weights could result in a stable spanning tree). On the other hand, the MV-BT based spanning trees incurred the least diameter (a measure of the delay incurred in receiving a broadcast message at all the nodes in the network) among the three topologies.

As part of future work, we will attempt to improve the stability of MV-BT based spanning trees by incorporating the relative velocity between the two end nodes and the difference in the direction of movement of the two end nodes of an edge as part of the weight of the edge. We will also evaluate the performance of these broadcast topologies in heterogeneous networks with the nodes operating under different transmission ranges.
5 Acknowledgments

The work leading to this paper was partly funded through the U.S. National Science Foundation (NSF) grants DUE-0941959 and CNS-0851646. The views and conclusions contained in this paper are those of the authors and do not represent the official policies, either expressed or implied, of the funding agency.

References