Interference-Aware Mesh Multicast for Wireless Multihop Networks

Daniel Lertpratchya
d.lertpratchya@gatech.edu

Douglas M. Blough
doug.blough@ece.gatech.edu

George F. Riley
riley@ece.gatech.edu

School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, Georgia 30332

ABSTRACT

In this paper, we consider the problem of building mesh-based multicast routing structures that account for the impact of interference in wireless multihop networks. Our analysis is based on the most accurate known interference model, namely the physical interference model. We first analyze interference-aware mesh structures that augment individual paths in a multicast tree. Based on this analysis, we propose two interference-aware multicast mesh routing structures, which extend an interference-aware Steiner multicast tree in two different ways to form interference-aware meshes. We evaluate the performances of our proposed interference-aware multicast mesh structures in wireless networks where wireless links are bursty and nodes can be faulty. Under these conditions, we show that our proposed algorithms provide up to 80% increase in goodput over existing tree-based multicast routing structures and up to 45% increase in goodput over existing mesh-based multicast routing structures.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless communication*

General Terms

Algorithms

Keywords

Multicast routing, Wireless interference

1. INTRODUCTION

In multicast, a single message is delivered to a group of destinations in a network. This problem has been studied for both wired and wireless networks. A survey of multicast protocols for ad hoc networks can be found in [3]. A major limitation of research in this area is that the vast majority of works ignore interference. The few works that do consider interference use inaccurate models.

MSWiM'14, September 21–26, 2014, Montreal, QC, Canada.
Copyright 2014 ACM 978-1-4503-3030-5/14/09 ...$15.00.
http://dx.doi.org/10.1145/2641798.2641836.

Multicast routing approaches can be classified into three main categories: structure-less (e.g. [6, 9]), tree-based (e.g. [4, 10]), and mesh-based (e.g. [1, 7]). Tree structures provide simple and cost effective routing infrastructures at the cost of robustness since there exists exactly one path between the source and each destination.

One possible solution to improve robustness is to use a mesh as the underlying routing structure instead of a tree. A mesh is a connected graph where there is more than one path from a multicast source to each multicast destination. These extra paths can deliver multicast packets to the destinations if the transmissions on other paths have failed. However, most of the mesh-based multicast protocols are concerned only with the problem of building and maintaining a multicast mesh efficiently but they ignore interference.

A few studies have been done on theoretical aspects of multicast mesh structures [11, 12]. Zhao and others [11] proposed four heuristics to build a resilient multicast mesh structure. Two heuristics, NDT and RNDT, build a multicast mesh by merging two node-disjoint MNT multicast trees [8] to form a multicast mesh. The other two heuristics, SDM and MDM, build a multicast mesh by finding a pair of node-disjoint shortest paths from the source node to each multicast receiver, then merging all the node-disjoint paths to form a multicast mesh. The proposed heuristics, however, do not take interference into account.

In this paper, we propose two interference-aware mesh multicast algorithms. Interference-aware multicast meshes are built by extending the interference-aware multicast tree [4]. The first algorithm creates a mesh by creating two redundant paths for each overlay link on the overlay multicast tree. The second algorithm uses Delaunay triangulation to build a multicast overlay mesh. We evaluate the performances of our proposed multicast mesh structures through simulation where link failure and node failure may occur. Simulation results show that our proposed multicast mesh structures provide up to 80% increase in goodput over existing tree-based routing structures and up to 45% increase over existing mesh-based routing structures. These results also show that our proposed interference-aware mesh routing structures are robust to the burstiness of wireless links.

2. SYSTEM MODEL AND PROBLEM FORMULATION

We consider a communication graph G, where $V(G)$ is a set of all wireless nodes and $E(G)$ is a set of edges. An edge $(u, v) \in E(G)$ if and only if $d(u, v) \leq r_t$, where $d(u, v)$ is the Euclidean distance between nodes u and v and r_t is the maximum transmission range. We are given a multicast source $s \in V(G)$ and a set of multicast destinations $M \subset V(G)$. The problem is to find a communication graph H where H is a connected subgraph of G and $M \cup \{s\} \subset V(H)$. As previously shown, the benefit of considering interference when building a multicast tree can be achieved even without explicit transmission scheduling [4]. Our goal is to find H that achieves the highest multicast packet reception ratio (MPRR), which is defined as the average packet reception ratio over all multicast destinations.

We adopt the classical model for radio signal propagation, which is referred to as the log-distance propagation loss model. The radio signal strength at a distance d from the transmitter is given by $\frac{P_t}{d^\alpha}$, where P_t is the transmission power and α is the path loss coefficient. We assume that all nodes use the same transmission power and they are not equipped with interference cancellation capabilities. We consider the physical interference (PI) model [2]. In the PI model, interference from all concurrent transmitters in the network is factored into the signal-to-interference ratio (SIR) value at the receiver. The transmission will be correctly received if and only if the SIR value at the receiver exceeds the SIR threshold ($\mathrm{SIR}_{\min}$).

To assist the analysis, we consider an ideal network where node density is infinite so we are able to pick nodes that satisfy the analysis when building multicast routing structures.

3. INTERFERENCE-AWARE MESH ROUTING STRUCTURES

In this section, we present algorithms to build interference-aware multicast mesh structures. The algorithms build a mesh by creating redundant paths on an interference-aware Steiner tree (IAST) [4] to form an interference-aware multicast mesh. The goal of the algorithms is to reduce the impact of interference among the paths in the mesh. before presenting algorithms to build interference-aware multicast meshes, we provide a quick review of the interference-aware Steiner tree (IAST).

3.1 Interference-aware Steiner Tree (IAST)

The high level idea of the interference-aware Steiner tree is as follows. Given nodes that must be connected in a multicast tree (the source node and all the destination nodes), the algorithm first identifies how these nodes should be connected in a tree. To accomplish this, the algorithm uses a Euclidean Steiner Tree approximation algorithm to build a Steiner tree, using $M \cup \{s\}$ as input. The Euclidean Steiner tree approximation algorithm returns a Steiner tree T where $V(T)$ is a set of nodes in the Steiner tree and $E(T)$ is a set of edges in the Steiner tree. The returned Steiner tree, also called an overlay tree, shows the "big picture" of connections between nodes in the tree. An edge between two nodes in the overlay tree suggests that the two nodes should be connected by a path in the multicast tree.

3.2 Overlay link extension algorithm

Our first tree extension algorithm is called the overlay link extension algorithm (OLE). Given an overlay tree T, the idea of our first multicast mesh algorithm is to create two redundant paths for each edge in the overlay tree. For each edge $(t_a, t_b) \in E(T)$, OLE creates two redundant paths between t_a and t_b. OLE places two shadow nodes u and v

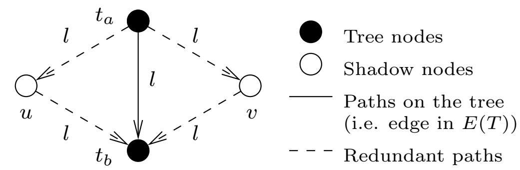
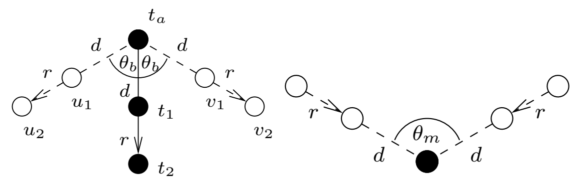


Figure 1: Overlay link extension algorithm.



(a) a branching point (b) a merging point

Figure 2: Two scenarios to be considered for OLE.

and creates two paths between t_a and t_b, one through each shadow node. Let $d(t_a, t_b) = l$, OLE places u and v such that $d(t_a, u) = d(t_a, v) = d(u, t_b) = d(v, t_b) = l$. The general idea of our first algorithm is illustrated in Figure 1.

By introducing two redundant paths between t_a and t_b, we have created one mesh branching point at t_a and one mesh merging point at t_b where the two redundant paths merge to t_b. Next, we determine optimal structures involving the branching point and the merging point in an ideal network.

3.2.1 *Mesh branching nodes*

Consider a mesh branching node t_a that branches into three nodes – one node on the overlay tree (t_1) and two nodes on the redundant paths (u_1 and v_1) as shown in Figure 2(a). Let $d = d(t_a, t_1) = d(t_a, u_1) = d(t_a, v_1)$ and $r = d(t_1, t_2) = d(u_1, u_2) = d(v_1, v_2)$. The transmission from t_a is done by broadcasting to all three children. Assuming that all three nodes successfully received the packet from t_a, the three nodes forward the packet to their next hops at the same time. Our goal is to find r such that the three transmissions will be successful. Among the three receivers, the node t_2 on the overlay tree experiences the largest interference. The total interference at t_2 is given by

$$\frac{2P_t}{(d^2 + dr + r^2)^{\alpha/2}}.$$

Combining the received signal strength and the interference, we set SIR to $\mathrm{SIR}_{\min}$ and convert units to decibel, we get $r = b \cdot d$ where

$$b = \frac{1 + \sqrt{1 + 4(10^{\frac{10\log 2 + \mathrm{SIR}^{\mathrm{dB}}_{\min}}{5\alpha}} - 1)}}{2(10^{\frac{10\log 2 + \mathrm{SIR}^{\mathrm{dB}}_{\min}}{5\alpha}} - 1)}.$$

The result shows that the distance between the transmitting nodes and the receiving nodes is proportional to the distance between the transmitting node and the mesh branching node. The distance grows as the transmitting node gets farther away from the mesh branching node until the distance reaches the maximum transmission range.

3.2.2 *Mesh merging nodes*

Next, we consider a mesh merging node in Figure 2(b) where the redundant paths finally merge back to the node on the overlay tree. Since the length of the tree path is l and the length of the redundant path is $2l$, the transmissions on the tree path will have already arrived at the merging node when the two redundant paths converge. Applying similar analysis as the mesh branching nodes, we get $r = m \cdot d$ where

$$m = \frac{3 + \sqrt{12 \cdot 10^{\frac{\text{SIR}_{\text{min}}^{\text{dB}}}{5\alpha}} - 3}}{2(10^{\frac{\text{SIR}_{\text{min}}^{\text{dB}}}{5\alpha}} - 1)}.$$

Here, the distance between nodes must shorten as the nodes get closer to the merging node. However, the last transmissions to the merging node cannot take place at the same time. The transmissions to the merging node may need to be scheduled to avoid collision.

After the overlay mesh in an ideal network is formed, the algorithm builds a multicast mesh routing structure using the overlay mesh as a guideline. Since the real network is finite, it is not possible to find nodes that are exactly at the shadow nodes' locations. The algorithm first finds a node that is nearest to a shadow node location and assigns the selected node as the shadow node.

Next, the algorithm builds a mesh by connecting nodes with paths, using the analyses for the mesh branching nodes and mesh merging nodes to select nodes on the paths. Again, since the real network is finite, finding nodes that completely satisfy the analyses is not possible. To solve this problem, we use a scaling factor (f) [4]. Let the distance from the analysis in an ideal network be r_i, the algorithm uses the distance $r_i^* = f \cdot r_i$, where $0 < f \leq 1$, when selecting nodes on the path. The purpose of the scaling factor is to account for the imperfect choices of nodes in a finite network.

One advantage of the OLE algorithm is that the analyses can be used as guidelines to select nodes on the final multicast mesh. However, the OLE algorithm creates two redundant paths for all overlay links that can result in overlapping redundant paths.

3.3 Delaunay mesh extension algorithm

In this section, we propose our second algorithm to extend a multicast tree to form a mesh, called Delaunay mesh extension algorithm (DME). Our goal in designing the second algorithm is to take the *overall* tree structure into account when building a multicast mesh.

3.3.1 Basic Delaunay mesh extension

The idea of the Delaunay mesh extension algorithm is to use Delaunay triangulation on the set of overlay nodes in the overlay multicast tree to identify the positions of the mesh nodes (also called Delaunay nodes). Given a Delaunay triangulation, the algorithm identifies a center point of each triangle as a position of a potential mesh node. The algorithm creates redundant paths between a potential mesh node and each of its corresponding nodes in the Delaunay triangulation if the overlay node is not a Steiner node. We do not create a redundant path between the potential mesh node and a Steiner node since a Steiner node is not a source or a receiver in the multicast group and does not need to be protected by a redundant path.

One advantage of the DME algorithm is that the redundant paths will not overlap each other. However, one drawback of the DME algorithm is that the analysis cannot be applied directly like the overlay link extension since the lengths of the redundant paths are not identical. To solve this problem, we again use a scaling factor to scale down the maximum distance between two nodes when building a multicast mesh. The distance between two nodes is fixed to $r = f \cdot r_t$ where $0 < f \leq 1$. In other words, DME algorithm uses a fixed, shortened distance when building a multicast mesh instead of using different routing strategies for branching nodes and merging nodes.

Even though the DME algorithm does not create overlapping redundant paths, it is still possible for two Delaunay nodes to be located close together. To solve this problem, we propose a variation of the DME algorithm next.

3.3.2 Delaunay mesh extension with nodes merging

The goal of the Delaunay mesh extension with nodes merging algorithm (DME-merge) is to merge two Delaunay nodes that are located closer than a given distance into one node.

The algorithm takes the mesh structure from the DME algorithm and repeatedly combines two Delaunay nodes if they are separated by a distance smaller than the given `merging_distance` by placing a new Delaunay node at the midpoint of the two nodes. The new Delaunay node connects to all the overlay nodes that were connected to the two Delaunay nodes. The merging process continues until no two Delaunay nodes satisfy the merging condition.

For comparison, we show an example of the three proposed interference-aware multicast routing structures in Figure 3.

4. PERFORMANCE EVALUATION

We evaluate the performances of our interference-aware multicast mesh structures through simulations. We evaluate the multicast routing structures under two major causes of network disconnection: link burstiness and node failure. A stochastic bursty wireless link model [5] is used for links. To simulate node failure, nodes in the network randomly drop multicast packets.

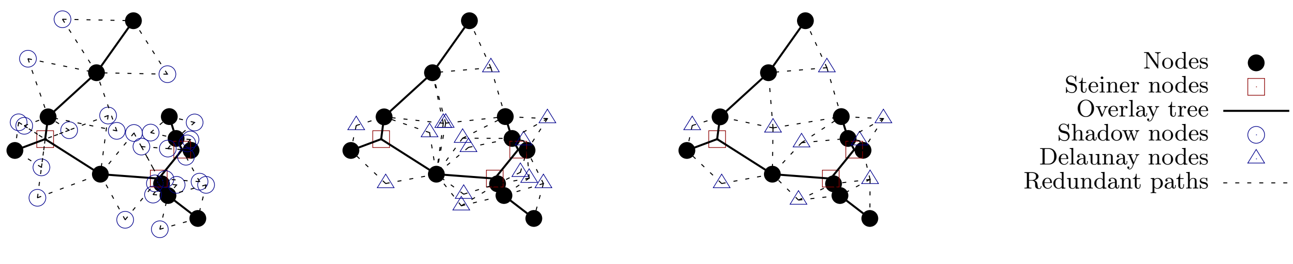
4.1 Simulation parameters and assumptions

We use ns-3.15 simulator to evaluate all algorithms. We use a physical model of 802.11g at the data rate of 6 Mbps. All nodes use the transmission power of 40 mW and thermal noise is computed at 290K. All wireless links are modeled with ideal bursty link model [5], unless stated otherwise. In all simulations, 2000 nodes were uniformly distributed in a deployment area of 1000 m by 1000 m. All results reported are reported using multicast packet reception ratio (MPRR) and are averaged from 100 simulations.

4.2 Scaling factor and merging distance

We first evaluate the scaling factor since it is a significant parameter affecting both OLE and DME. In this simulation, one source node is randomly selected as a multicast source. The source node sends multicast packets at the rate of 10 packets per second. Scaling factor was varied from 0.3 to 1.0 for both OLE and DME. We did not use the scaling factor below 0.3 since the network became disconnected in some simulations. Simulation results are reported in Figure 4.

As seen from Figure 4, the choice of scaling factor affects the performance of OLE and DME algorithms. If the scaling factor is too small, the extra nodes included create more interference that can outweigh the gain of spatial reuse. If the scaling factor is too large, the links are prone to interference from other concurrent transmissions. The optimal scaling



(a) OLE (b) DME (c) DME-merge

Figure 3: Examples of different interference-aware multicast routing structures.

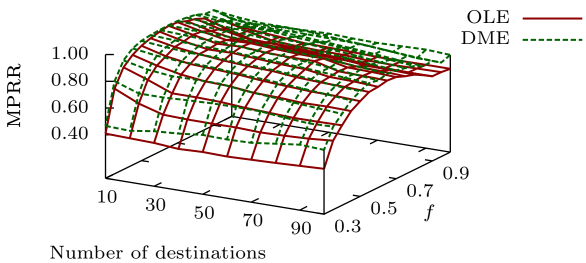


Figure 4: MPRR with varying scaling factors.

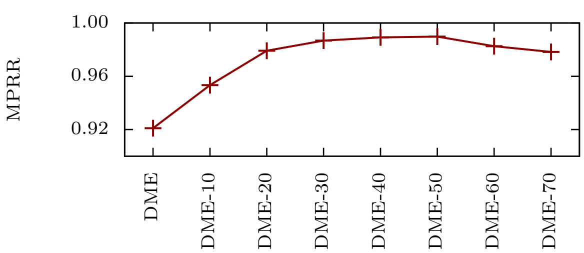


Figure 5: MPRR of DME-merge algorithm with different merging distance.

factor is also dependent on the number of multicast destinations in the network as the number of nodes in the multicast trees changes. Performances are quite stable across a fairly wide range of scaling factors for both OLE and DME, e.g. 0.5 to 0.7. Based on this analysis, we have set the scaling factor for both algorithms to 0.7.

Next, we evaluate the merging distance parameter of DME-merge algorithm. In this simulation, the merging distance of DME-merge was varied from 10 m to 70 m (DME-10 to DME-70). MPRR of DME-merge are reported in Figure 5.

Figure 5 confirms that the choice of merging distance affects the performance of DME-merge algorithm. When the merging distance is small, only a few Delaunay nodes are merged together, and the mesh structure of DME-merge is still similar to the mesh structure of DME. However, if the merging distance is large, DME-merge will aggressively merge Delaunay nodes. This aggressive merging can result in reduced performance. When DME-merge merges two Delaunay nodes, the resulting Delaunay node is responsible for all links of the original Delaunay nodes. Thus, the more that Delaunay nodes are merged, the more links the new Delaunay node must handle. In an extreme case, this can create a

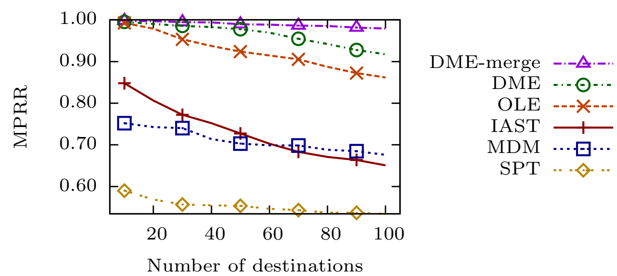


Figure 6: MPRR of different multicast routing structures.

Delaunay node that is connected to all other overlay nodes, while having only one incoming path to the Delaunay node.

Performance are quite stable across a fairly wide range of merging distances, e.g. 30 to 50. Based on these results, we have set the merging distance to 50 m.

4.3 Multicast routing with bursty wireless links

In this simulation, we evaluate the performances of different multicast routing structures when the wireless links exhibit bursty behavior. The number of multicast destinations was varied from 10 to 100. A single multicast source was randomly selected among the remaining nodes. The source node generates multicast packets at the rate of 10 packets per second for 600 seconds. We have implemented another mesh multicast routing structure called MDM for comparison [11]. Simulation results are reported in Figure 6.

As seen from Figure 6, the performances of different multicast routing structures vary. The shortest path tree has the lowest MPRR among all multicast routing structures. Since the shortest path tree does not consider interference when building a tree, it is more prone to interference than other structures. The tree structure is also vulnerable to even a single transmission failure as the tree will be disconnected.

The interference-aware Steiner tree provides improvement over the shortest path tree since IAST takes interference into account when building a tree. As a result, IAST is less prone to interference than the shortest path tree. Still, a single transmission failure will disconnect the multicast tree of IAST. MDM also provides improvement over the shortest path tree by including redundant paths to reach the destinations. However, since MDM does not take interference into account when building a mesh, it is still prone to transmission failure along the mesh.

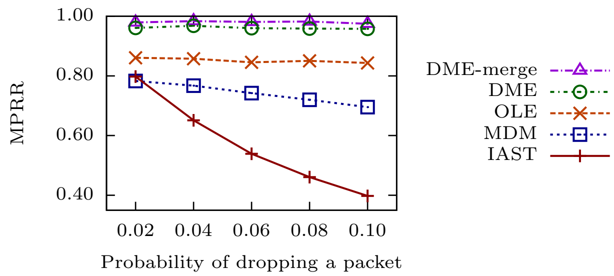


Figure 7: MPRR at varying faulty node probability.

OLE and DME algorithms have higher MPRR than other multicast routing structures, including our previously proposed interference-aware Steiner tree algorithm. The redundant paths allow the multicast packets to be delivered to the destinations even if some multicast packets were dropped on the way to the receivers. Moreover, the paths built by both OLE and DME algorithms are interference-aware, which makes the links less prone to transmission failure than other multicast mesh routing structures. Among our three mesh algorithms, Delaunay mesh extension with nodes merging provides the best MPRR.

4.4 Multicast routing with faulty nodes

In this set of simulations, we study another cause for disconnected graph – node failure. For simplicity, we assume that wireless links are not bursty in this study. To simulate a faulty node, each node randomly drops a multicast packet instead of forwarding the packet to the next node. The probability of dropping a multicast packet was varied from 0.02 to 0.10. The decision to drop the packet is made independently for each packet. The number of multicast destinations was kept constant at 50. MPRR of different multicast routing structures are reported in Figure 7.

As seen from Figure 7, the performances of most multicast routing structures drop as the failure probability increases. The performance drop is substantial for IAST since it relies on a tree as a multicast routing structure. A single faulty node along the tree will disconnect the subtree below the faulty node.

For MDM, the multicast packet reception ratio drops from about 0.80 to about 0.70. MDM relies on mesh structure, which makes it more robust when a few nodes are faulty. However, the number of redundant paths of MDM is not large enough to handle a large number of faulty nodes, which results in a drop in multicast packet reception ratio when the failure probability is high.

Our proposed mesh multicast routing structures can withstand a larger number of faulty nodes than MDM and IAST as can be seen by the almost constant multicast packet reception ratios even at a high fault probability. The extra paths included by OLE, DME, and DME-merge make them more robust to faulty nodes than other routing structures.

5. CONCLUSION

In this paper, we have proposed two algorithms to extend the interference-aware Steiner multicast tree to create an interference-aware multicast mesh. The main idea of both algorithms is to include a set of redundant paths that are connected back to the overlay tree to form an overlay mesh. The algorithms build the actual multicast mesh structure using the overlay mesh as a guideline. We have evaluated our proposed algorithms in three different settings and showed that our algorithms provide higher multicast packet reception ratios than other multicast routing structures that do not consider interference.

Acknowledgements

This research was supported in part by the National Science Foundation under Grants CNS-1017248 and CNS-1319455.

6. REFERENCES

[1] I. Er and W. Seah. Distributed Steiner-like multicast path setup for mesh-based multicast routing in ad hoc networks. In *IEEE Intl. Conf. Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006.*, volume 2, pages 192–197, June 2006.

[2] P. Gupta and P. Kumar. The capacity of wireless networks. *IEEE Trans. Information Theory*, 46(2):388–404, Mar 2000.

[3] L. Junhai, Y. Danxia, X. Liu, and F. Mingyu. A survey of multicast routing protocols for mobile ad-hoc networks. *IEEE Communications Surveys Tutorials*, 11(1):78–91, First 2009.

[4] D. Lertpratchya, D. M. Blough, and G. F. Riley. Interference-aware multicast for wireless multihop networks. In *IEEE Wireless Communications and Networking Conference*, WCNC 2014, April 2014.

[5] D. Lertpratchya, G. F. Riley, and D. M. Blough. Simulating frame-level bursty links in wireless networks. In *Proc. 7th Intl. ICST Conf. Simulation Tools and Techniques*, SimuTools '14, March 2014.

[6] J.-S. Park, M. Gerla, D. Lun, Y. Yi, and M. Medard. Codecast: a network-coding-based ad hoc multicast protocol. *IEEE Wireless Communications*, 13(5):76–81, October 2006.

[7] S. Park and D. Park. Adaptive core multicast routing protocol. *Wireless Networks*, 10(1):53–60, 2004.

[8] P. Ruiz and A. Gomez-Skarmeta. Approximating optimal multicast trees in wireless multihop networks. In *Proc. 10th IEEE Symp. Computers and Communications*, pages 686–691, June 2005.

[9] J. Sanchez, P. Ruiz, and I. Stojmenovic. GMR: Geographic multicast routing for wireless sensor networks. In *3rd Annual IEEE Communications Society Conf. Sensor and Ad Hoc Communications and Networks*, volume 1, pages 20–29, Sept 2006.

[10] J. Yi and C. Poellabauer. Real-time multicast for wireless multihop networks. *Computers & Electrical Engineering*, 36(2):313 – 327, 2010. Wireless ad hoc, Sensor and Mesh Networks.

[11] X. Zhao, C. T. Chou, J. Guo, and S. Jha. Protecting multicast sessions in wireless mesh networks. In *Proc. 31st Annual IEEE Intl. Conf. Local Computer Networks*, LCN 2006, pages 467–474, Nov 2006.

[12] Y. Zheng, U. T. Nguyen, and H. L. Nguyen. Data overhead impact of multipath routing for multicast in wireless mesh networks. In *2012 Third FTRA Intl. Conf. Mobile, Ubiquitous, and Intelligent Computing (MUSIC)*, pages 154–157, June 2012.