Volume 2, Issue 2, February 2012 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Low Bandwidth utilization in Wireless Mesh Networks

[1]P.Balakumar
Research Scholar ,
Dept Of CSE
Bharath university

[2]Dr.M.Rajani,
Director Research,
Bharath university

Abstract-We study the problem of computing multicast trees with minimal bandwidth consumption in multi-hop wireless mesh networks. For wired networks, this problem is known as the Steiner tree problem, and it has been widely studied before. We demonstrate in this paper, that for multihop wireless mesh networks, a Steiner tree does not offer the minimal bandwidth consumption, because it neglects the wireless multicast advantage. Thus, we re-formulate the problem in terms of minimizing the number of transmissions, rather than the edge cost of multicast trees. We show that the new problem is also NP-complete and we propose heuristics to compute good approximations for such bandwidth-optimal trees. Our simulation results show that the proposed heuristics offerr a lower bandwidth consumption compared with Steiner trees.

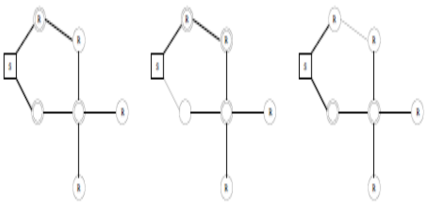
1 Introduction and Motivation

A wireless multihop network consists of a set of nodes which are equipped with wireless interfaces. Nodes which are not able to communicate directly use multihop paths using other intermediate nodes in the network as relays. When the nodes are free to move, these networks are usually known as”mobile ad hoc net-works”. We focus on this paper in static multihop wireless networks, also known as”mesh networks”. These networks have recently received a lot of attention in the research community, and they are also gaining momentum as a cheap and easy way for mobile operators to expand their coverage and quickly react to temporary demands.

In addition, IP multicast is one of the areas which are expected to play key role in future mobile and wireless scenarios. Key to this is the fact that many of the future services that operators and service providers for see are bandwidth-avid, and they are strongly based on many-to-many interactions. These services require an efficient underlying support of multicast communications when deployed over multihop extensions where bandwidth may become a scarce resource.

The problem of the efficient distribution of tarffic from a set of senders to a group of receivers in a datagram network was already studied by Dearing in the late 80’s. Several multicast routing protocols like DVMRP, MOSPF, CBT and PIM) have been proposed for IP multicast routing in fixed networks. These protocols have not been usually considered in mobile ad hoc.

In this paper we show that the Steiner tree does not always give an optimal solution. Additional contributions of this paper are the demonstration that the problem of minimizing the cost of a multicast tree in a wireless mesh network is also NP-complete, and the proposal of enhanced heuristics to approximate such optimal trees, which we call minimal data overhead trees. Our simulation



(a) Shortest Path Tree, 4 Tx/packet

(b) Steiner Tree, 4 Tx/packet

(c) Min. Data Overhead, 3 Tx/packet

Fig. 1. Differences in cost for several multicast trees over the same ad hoc network results show that the proposed heuristics produce multicast trees with lower bandwidth consumption that previous heuristics for Steiner trees over a variety of scenarios. In addition, they offer a huge reduction in the cost compared to the shortest path trees used by most of the ad hoc multicast routing protocols proposed so far.

The remainder of the paper is organized as follows: section 2 describes our network model, formulates the problem and shows that it is NP-complete. The description of the proposed algorithm is given in section 3. In section 4 we explain our simulation results. Finally, section 5 provides some discussion and conclusions.

2 Network Model and Problem Formulation

2.1 Network model

We represent the ad hoc network as an undirected graph G(V,E) where V is the set of vertices and E is the set of edges. We assume that the network is two dimensional (every node v 2 V is embedded in the plane) and mobile nodes are represented by vertices of the graph. Each node v 2 V has a transmission range r. Let dist(v1, v2) be the distance between two vertices v1, v2 2 V . An edge between two nodes v1, v2 2 V exists iff dist(v1, v2) _ r (i.e. v1 and v2 are able to communicate directly). In wireless mobile ad hoc networks some links may be unidirectional due to different transmission ranges. However, given that lower layers can detect and hide those unidirectional links to the network layer, we only consider bidirectional links. That is, (v1, v2) 2 E iff (v2, v1) 2 E.

3.Proposed Algorithms

Given the NP-completeness of the problem, within the next subsections we describe two heuristic algorithms to approximate minimal data-overhead multicast trees. As we learned from the demonstration of theorem 2, the best approach to reduce the data overhead is reducing the number of forwarding nodes, while increasing the number of leaf nodes. The two heuristics presented below try to achieve that trade-off.

3.1 Greedy-based heuristic algorithm

The first proposed algorithm is suited for centralized wireless mesh networks, in which the topology can be known by a single node, which computes the multicast tree. Inspired on the results from theorem 2, this algorithm systematically builds different cost-effective sub trees. The cost-effectiveness refers to the fact that a node v is selected to be a forwarding node only if it covers two or more nodes. The algorithm shown in algorithm 1, starts by initializing the nodes to cover ('aux') to all the sources except those already covered by the source s. Initially the set of forwarding nodes ('MF') is empty. After the initialization, the algorithm repeats the process of building a cost-effective tree, starting with the node v which covers more nodes in 'aux'. Then, v is inserted into the set of forwarding nodes (MF) and it becomes a node to cover. In addition, the receivers covered by v (Cov(v)) are removed from the list of nodes to cover denoted by 'aux'.

This process is repeated until all the nodes are covered, or it is not possible to find more cost-effective sub trees. In the latter case, the different sub trees are connected by a Steiner tree among their roots, which are in the list 'aux' (i.e. among the nodes which are not covered yet). For doing that one can use any Steiner tree heuristic. In our simulations we use the MST heuristic for simplicity.

Algorithm 1 Greedy minimal data overhead algorithm

```
MF   _/ _mcast − forwarders _ /
V   V - {s}
aux   R-Cov(s) + {s} / _ nodes − to − cover _ /
repeat
node   argmaxv2V (|Cov(v)|) s.t. Cov(v)_2
aux   aux-Cov(v)+{v}
```

```
    V   V-{v}
    MF   MF + {v}
until aux = _ or node = null
if V!=_ then
    Build Steiner tree among nodes in aux
end if
```

3.2 Distributed version of the algorithm

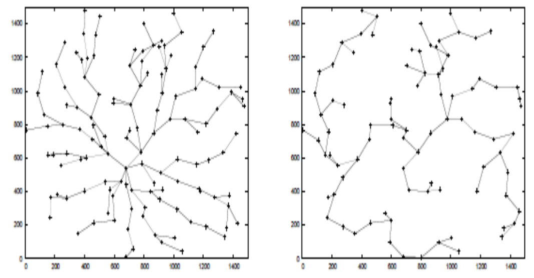
The previous algorithm may be useful for some kind of networks; however a distributed approach is much more appealing for the vast majority of scenarios. In this section we present a slightly different version of the previous algorithm, being able to be run in a distributed way. The previous protocol consists of two different parts: (i) construction of cost-efficient sub trees, and (ii) building a Steiner tree among the roots of the sub trees.

To build a Steiner tree among the roots of the sub trees, we assumed in the previous protocol the utilization of the MST heuristic. However, this is a centralized heuristic consisting of two different phases. Firstly, the algorithm builds the metric closure for the receivers on the whole graph, and then, a minimum spanning tree (MST) is computed on the metric closure. Finally, each edge in the MST is substituted by the shortest path tree (in the original graph) between the to nodes connected by that edge. Unfortunately, the metric closure of a graph is hard to build in a distributed way. However, we can approximate such an MST heuristic with the simple, yet powerful, algorithm presented in algorithm 2. The source, or the root of the sub tree in which the source is (called source-root) will start flooding a route request message (RREQ). Intermediate nodes, when propagating that message will increase the hop count. When the RREQ is received by a root of a sub tree, it sends a route reply (RREP) back through the path which reported the lowest hop count. Those nodes in that path are selected as multicast forwarders (MF). In addition, a root of a sub tree, when propagating the RREQ will reset the hop count field. This is what makes the process very similar to the computation of the MST on the metric closure. In fact, we achieve the same effect, which is that each root of the sub trees, will add to the Steiner tree the path from itself to the source-root, or the nearest root of a sub tree. The way in which the algorithm is executed from the source-root to the other nodes guarantees that the obtained tree is connected.

Algorithm 2 Distributed approximation of MST heuristic

```
if thisnode.id = source − root then
    Send RREQ with RREQ.hopcount=0
end if
if rcvd non duplicate RREQ with better hopcount then
    prevhop   RREQ.sender
    RREP.nexthop   prevhop
    RREQ.sender   thisnode.id
    if thisnode.isroot then
        send(RREP)
        RREQ.hopcount   0
    else
        RREQ.hopcount++;
    end if
    send(RREQ)
end if
if received RREP and RREP.nexthop = thisnode.id then
    Activate MF FLAG
    RREP.nexthop   prevhop
    send(RREP)
end if
```

The second part of the algorithm to make distributed is the creation of the cost-effective sub trees. However, this part is much simpler and can be done locally with just a few messages. Receivers flood a Sub tree Join (ST JOIN) message only to its 1-hop neighbors indicating the multicast group to join. These neighbors answer with a Sub tree Join Ack (ST ACK) indicating the number of receivers they cover. This information is known locally by just counting the number of (ST JOIN) messages received. Finally, receivers send again a Sub tree Join Activation (ST JOIN ACT) message including their selected root, which is the neighbor which covers a higher number of receivers. This is also known locally from the information in the (ST ACK). Those nodes which are selected by any receiver, repeat the process acting as receivers. Nodes which already selected a root do not answer this time to ST JOIN messages.



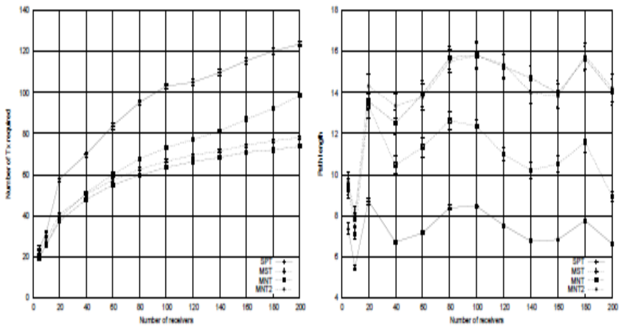
(a) SPT (b) MST

Fig. 2. Shape of example trees produced by heuristics

4 Simulation Results

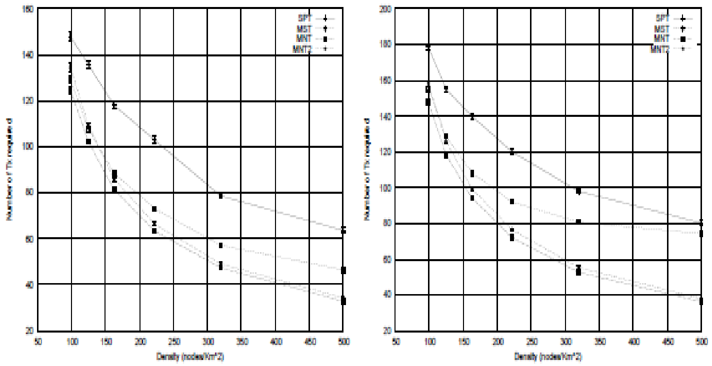
In order to assess the effectiveness of our proposed algorithms we have simulated them under different conditions. The algorithms that we have simulated are the two proposed approaches as well as the MST heuristic to approximate Steiner trees. In addition, we also simulated the shortest path tree algorithm, which is the one which is used by most multihop multicast routing protocols proposed to date.

As in many similar papers, we do not consider mobility in our simulations, because we are dealing with wireless mesh networks. In mobile ad hoc networks, changes in the topology make useless to approximate optimal solutions, which may become suboptimal within a few seconds.



(a) Number of transmissions (b) Mean path length

Fig. 3. Number of Tx and mean path length at increasing number of receivers.



(a) 100 receivers (b) 180 receivers

Fig. 4. Number of Tx with varying network density

Conclusions and discussion

As we have shown, the generally considered minimal cost multicast tree (Steiner tree) does not offer an optimal solution in multihop wireless networks. The problem is that the original Steiner tree problem formulation does not account for the reduction in bandwidth that can be achieved in a broadcast medium. Given those limitations we re-formulate the problem in terms of minimizing the number of transmissions required to send a packet from a multicast source to all the receivers in the group.

We have shown that this formulation is adequate for multihop wireless networks, and we have also demonstrated that this problem is NP-complete. So, we have introduced two new heuristic algorithms to deal with the problem of optimizing multicast trees in wireless mesh networks. Our simulation results show that the proposed heuristics manage to beat the Steiner tree MST heuristic over a variety of scenarios and network densities.

In particular, our results show that the higher the density of the network, the higher are the performance gains introduced by our heuristics compared to the other approaches. These results seem very promising as a possible future direction to address similar issues in sensor networks in which the network topology is generally very dense, and reverse multicast trees are very common as a mechanism to gather information from the sensor network.

References

1. S. Deering, "Multicast Routing in a Datagram Internetwork," Ph.D. Thesis, Electrical Engineering Dept., Stanford University, Dec. 1991.

2. S.-E. Deering and D.-R. Cheriton, "Multicast Routing in datagram internetworks and extended LANs," Transactions on Computer Systems, vol.8, no.2, May 1990, pp. 85–110.

3. J. Moy, "Multicast routing extensions for OSPF," Computer communications of the ACM, vol.37, no.8, August 1994, pp.61–66.

4. T. Ballardie, P. Francis and J. Crowcroft, "Core Based Trees (CBT) – An archi- tecture for scalable inter-domain multicast routing, " Proc. of ACM SIGCOMM'93, San Francisco, CA, October 1993, pp.85–95.

5. S. Deering, D.-L. Estrin, D. Farinacci, V. Jacobson, C.-G. Liu and L. Wei, "The PIM architecture for wide-area multicast routing," IEEE/ACM Transactions on Networking, vol.4, no.2, April 1996, pp. 153–162.

6. C. Cordeiro, H. Gossain and D. Agrawal, "Multicast over Wireless Mobile Ad Hoc Networks: Present and Future Directions" IEEE Network, no. 1, Jan 2003, pp. 52–59.

7. R.-M. Karp, "Reducibility among combinatorial problems," In Complexity of com- puter computations, Plenum Press, New York, 1975, pp.85–103.

8. B.-M. Waxman, "Routing of Multipoint Connections," IEEE Journal on Selected Areas in Communications, vol. 6, no. 9, December 1998, pp. 1617–1622.

9. L. Kou, G. Markowsky, and L. Berman, "A fast algorithm for Steiner trees," Acta Informatica, no. 15, vol. 2, 1981, pp. 141–145.

10. J. Plesnik, "The complexity of designing a network with minimum diameter," Networks, no. 11, 1981, pp. 77–85.

11. A. Zelikovsky, "An 11/6-approximation algorithm for the network Steiner prob- lem," Algorithmica, no. 9, 1993, pp.463–470.

12. S. Rajagopalan and V. V. Vazirani. "On the bidirected cut relaxation for the metric Steiner tree problem," in Proceedings of the 10th Annual ACM-SIAM Symposium on Discrete Algorithms, 1999, pp. 742–751.

13. S. Even, "Graph Algorithms," Computer Science Press, 1979, pp. 204–209.

14. H. Lim and C. Kim, "Multicast Tree Construction and Flooding in Wireless Ad Hoc Networks," Proceedings of the 3rd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems, Boston, MA, USA. August, 2000, pp. 61–68.