Available online at www.sciencedirect.com

ScienceDirect

Computer Communications 31 (2008) 1385–1399

computer communications

www.elsevier.com/locate/comcom

On multicast routing in wireless mesh networks

Uyen Trang Nguyen *

Department of Computer Science and Engineering, York University, 4700 Keele Street, Toronto, Ont., Canada M3J 1P3

Abstract

There are two fundamental approaches to multicast routing: shortest path trees (SPTs) and minimum cost trees (MCTs). The SPT algorithms minimize the distance (or cost) from the sender to each receiver, while the MCT algorithms such as minimum Steiner trees (MSTs) minimize the overall edge cost of the multicast tree. In wireless multi-hop networks, the tree cost can be redefined to exploit the wireless broadcast advantage: a minimum cost tree is one which connects sources and receivers by issuing a minimum number of transmissions (MNT). Among the different approaches, SPT is the more commonly used method for multicast routing in the Internet. The MNT approach was originally considered for energy-constrained wireless networks such as sensor and mobile ad-hoc networks. It is not clear how the different types of trees compare when used in WMNs. In this paper, we present a simulation-based performance comparison of SPTs, MSTs and MNT trees in WMNs using most concerned performance metrics such as packet delivery ratio, throughput, end-to-end delay, delay jitter and multicast traffic overheads. Based on the experimental results, we provide insights into the performance of multicast routing algorithms in WMNs and recommendations for suitable routing approaches.

Keywords: Multicast routing; Shortest path trees; Minimum cost trees; Wireless mesh networks; Wireless broadcast advantage

1. Introduction

Wireless mesh networking is an emerging technology that supports many important applications such as Internet access provisioning in rural areas, municipal and metropolitan networking for emergency and disaster recovery, security surveillance, and information services in public transportation systems [1]. Major components of a wireless mesh network (WMN) include wireless mesh routers, wireless hosts (e.g., PCs, laptops, PDAs, and cell phones), and access points (or gateways) that act both as Internet routers and wireless mesh routers. The mesh routers in a WMN provides multi-hop connectivity from one host to another, or to the Internet via the access points. The routers automatically establish and maintain mesh connectivity among themselves, making WMNs dynamically self-organized and self-configured networks. This feature brings many benefits to WMNs such as low installation cost, large-scale deployment, reliability, and self-management.

Multicast [2] is a form of communication that delivers information from a source to a set of destinations simultaneously in an efficient manner; the messages are delivered over each link of the network only once and only duplicated at branch points, where the links to the destinations split. Important applications of multicast include distribution of financial data, billing records, software, and newspapers; audio/video conferencing; distance education; and distributed interactive games. Although multicast is required to support many important applications, research on multicasting in WMNs is still in its infancy. In this paper, we address one of the most essential issues of multicast in WMNs – routing.

There are two fundamental multicast routing approaches: shortest path trees (SPTs) and minimum cost trees (MCTs). The goal of SPT algorithms is to construct a tree rooted at the sender and spanning all the receivers such that the distance between the sender and each receiver along the tree is minimum. As a result, the SPT algorithms normally minimize the *end-to-end delay* as well. The two

---

* Tel.: +1 416 736 2100; fax: +1 416 736 5872.
E-mail address: utn@cse.yorku.ca

0140-3664/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2008.01.022

most commonly used algorithms for computing SPTs are due to Bellman–Ford and Dijkstra [3]. To compute a SPT, we apply the point-to-point shortest path algorithm repeatedly, once for each sender–receiver pair.

Unlike the SPT algorithms, which aim at minimizing the distance (or cost) from the sender to each receiver, the goal of MCT algorithms is to minimize the overall cost of the multicast tree. MCT algorithms for multicast routing are based on the minimum Steiner tree (MST) problem, which is NP-complete. Thus several heuristics have been proposed to compute approximate Steiner trees [2], e.g., the 2-approximation heuristic proposed by Kou et al. [4], and the 11/6-approximation algorithm by Zelikovsky [5].

The total cost of a Steiner tree is less than the total cost of a corresponding SPT, by definition of Steiner trees. However, the maximum distance between the sender and any receiver in a Steiner tree is typically longer than that in a SPT. This means that the average path length in a Steiner tree is more than that in a SPT.

Due to the complexity of computing Steiner trees in a distributed manner, the majority of the multicast routing protocols used in the Internet today are based on SPTs, such as Distance Vector Multicast Routing Protocol (DVMRP) and Multicast Open Shortest Path First (MOSPF) [2]. The reason is that SPTs are easy to implement and offer minimum end-to-end delay, a desirable quality of service parameter for most real-life multicast applications.

Recently, Ruiz and Gomez-Skarmeta explored the problem of multicast routing in wireless multi-hop networks in which nodes are static, e.g., WMNs [6]. The authors re-define the cost of a MCT by applying the *wireless broadcast advantage*: in a broadcast medium, the transmission of a multicast data packet from a given node to any number of its neighbors can be done with a single data transmission. Thus, in a wireless multi-hop network, the minimum cost tree is one which connects sources and receivers by issuing a minimum number of transmissions, rather than having a minimal edge cost as defined for traditional minimum Steiner trees (MSTs). In other words, the tree contains a *minimum number of multicast forwarding nodes*. In general, a tree with a minimum edge cost may not be one with a minimum number of transmissions [6].

A routing tree that exploits the wireless broadcast advantage serves two purposes: to minimize the energy consumption of nodes in the network, and to minimize the network bandwidth consumption. Both goals are critical design objectives for multicast routing in wireless networks whose nodes have limited power supply such as mobile ad-hoc and sensor networks. Although minimizing energy consumption is not a concern for routers in WMNs in most applications, minimizing bandwidth consumption remains an important goal.

Ruiz and Gomez-Skarmeta demonstrated that the problem of computing Minimum Number of transmissions Trees (MNTs) is NP-complete and proposed enhanced heuristics to approximate such trees [6]. They presented experimental results that show that their MNT heuristics offers the least number of transmissions compared with the MST and the SPT algorithms in most cases. On the other hand, the mean path lengths given by the MST and the MNT algorithms are longer than that by the SPT algorithm, as expected. Nevertheless, the presented experimental results did not indicate how the multicast groups perform in terms of packet loss rate (or packet delivery ratio) – the true performance measure of a multicast session – or end-to-end delay – an important performance metric for real-time multicast applications such as distribution of stock quotes, distributed interactive games and teleconferencing – or delay jitter – a critical metric for audio/video applications.

One could argue that SPTs are best in an environment where the network topology is unknown and the multicast members may be geographically distributed over a very large area, such as the Internet. In a WMN where the topology is usually known and the network size is much smaller (e.g., less than 500 nodes), MCTs such as MSTs or MNTs are no longer difficult to implement, and could potentially offer better performance because they typically consume less bandwidth than SPTs. On the other hand, MCT algorithms produce longer paths than SPT algorithms. In a wireless multi-hop network, the longer the path, the higher the probability that a packet will be lost due to packet collision or congestion, resulting in a throughput reduction. One could thus infer that SPTs could achieve higher throughput than MCTs. In short, it is not clear how the performance of SPTs and MCTs in a wireless multi-hop network compares.

In this paper, we present a simulation-based performance comparison of minimum cost trees and shortest path trees. Specifically we examine the following two minimum cost tree algorithms: the minimum Steiner tree (MST) heuristic proposed by Kou et al. [4], and the MNT heuristic by Ruiz and Gomez-Skarmeta [6]. Since both the Bellman–Ford and the Dijkstra's algorithms converge to the same solution under static conditions of topology and non-negative link costs, we use only one of them – in this case, the Dijkstra's algorithm – for computing SPTs. We focus on one-to-many multicast in this paper, which supports many important applications such as distributions of software, financial data, database updates and news; pay-per-view movies; IP television; and distance learning.

We compare the average path length and the number of forwarding nodes of the routing trees built by the MST, MNT and SPT algorithms. We measure packet delivery ratio, throughput, end-to-end delay and delay jitter of multicast receivers in shortest-path trees and minimum-cost trees. We also study the effects of data traffic generated by multicast nodes on the packet loss rates of unicast flows in the same network. Based on the experimental results, we provide insights into the performance of multicast routing algorithms in WMNs as well as recommendations for suitable routing approaches.

The remainder of this paper is structured as follows. We first describe our simulation setting and the performance metrics in Section 2. In Section 3, we present experimental results that compare the performance and traffic overheads of multicast trees constructed by the MST, MNT and SPT algorithms. We then discuss the results and give recommendations in Section 4. Related work is presented in Section 5. Finally, we outline our future work and conclude the paper in Section 6.

2. Experiment setting

Our experiments were carried out using QualNet [7], a software that provides scalable simulations of wireless networks and a commercial version of GloMoSim [8]. We implemented the Dijkstra's (SPT) algorithms as described in [16], the MST heuristic proposed by Kou et al. [4], and the centralized MNT heuristic by Ruiz and Gomez-Skarmeta [6]. Following are our performance metrics and simulation parameters.

2.1. Performance metrics

We use the following metrics to measure the performance of a multicast routing protocol:

- *Average multicast packet delivery ratio*. The packet delivery ratio (PDR) of a receiver is the number of data packets actually delivered to the receiver versus the number of data packets supposed to be received. The average PDR of a multicast group is the average of the PDRs of all the receivers in the group.
- *Average end-to-end delay*. The end-to-end delay of every packet received at every receiver is recorded; the average over all the packets received is then computed.
- *Average throughput*. The throughput is defined as the total amount of data a receiver actually receives divided by the time between receiving the first packet and the last packet. The average taken over all the receivers is the average throughput of the multicast group, assuming that each group has one sender.
- *Average delay jitter*. Delay jitter is the variation (difference) of the inter-arrival intervals from one packet received to the next packet received. The per-receiver delay jitter at a receiver is the sum of all the absolute values of delay jitters from the first packet received to the last packet received divided by the total number of packets received. The average delay jitter is the average of the per-receiver delay jitters taken over all the receivers.
- *Average unicast packet delivery ratio*. To measure the impacts of multicast data traffic on the PDRs of unicast flows in a network, we recorded the PDR of every unicast flow and then took the average over all the unicast flows.
- *Average path length*. The path length (or the hop count) is an indirect indicator of performance: in general, the longer a path, the higher the packet loss rate of a flow and the longer the end-to-end delay. The average path length is the average of the lengths of all source-to-destination paths in a multicast tree.
- *Number of forwarding nodes*. The number of forwarding nodes in a multicast tree is also an indirect indicator of performance: the lower the number, the less network bandwidth consumed by the multicast group (i.e., the wireless broadcast advantage). This affects the packet delivery ratios of the multicast group as well as of other flows in the network.

2.2. Simulation parameters

Our simulation models a medium-size network of 100 wireless routers uniformly distributed over a 2000 m × 2000 m area, and a large network of 300 wireless routers, over a 3000 m × 3000 m area. We will use the term "wireless router" and "node" interchangeably in this paper. The nodes are distributed uniformly over the subareas within a terrain, and the nodes within a sub-area are randomly placed in that sub-area. There are no network partitions throughout the simulation. The edge cost or the cost of each wireless link is assumed to be one.

The transmission power of the routers is set constant at 20 dBm; the data transmission rate at the physical layer is 11 Mbits/s; the transmission range of the wireless routers is 315 m, according to the specifications of wireless routers manufactured by TROPOS [10]. We use PHY802.11b at the physical layer. A two-ray propagation model [9] is used when the distance between two routers is 250 m or more; otherwise, a free space model is used to avoid the oscillation caused by the constructive and destructive combination of the two rays over short distances. The above distance threshold for switching between the two models is calculated by the QualNet software.

The MAC802.11 protocol with Distributed Coordination Function (DCF) is chosen as the medium access control protocol. We implement only one channel on each wireless link as explained in Section 2.3. Unicast flows use Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) in combination with RTS/CTS/DATA/ACK (request to send/clear to send/data/acknowledgment) exchanges. Multicast flows use only CSMA/CA without RTS, CTS or ACK, at both branch point (one-to-many) and non branch point (one-to-one) nodes in a routing tree (see Section 2.3 for a detailed explanation).

The data packet size excluding the header size is 512 bytes. The size of the queue at every node is 50 Kbytes. The packets in a queue are scheduled on a first-in-first-out basis. We did not implement flow or congestion control in order to test the network performance under very high loads.

Each multicast group has one sender. In each experiment, the sender of a multicast group transmits at a *constant bit rate*, from 10 to 80 packets/s. The number of receivers (or the group size) is varied from 20 to 80, unless otherwise stated. We assume that each sender or receiver is connected to a *different* wireless router. (In practice, there can be many hosts communicating with a wireless router, e.g., to form a wireless local area network.) The sender and the receivers of a multicast group were selected randomly, and the same sender and receivers and the same network configuration were used for all three algorithms (MST, MNT and SPT) in order to obtain a fair comparison. All receivers joined a multicast group at the beginning and stayed until the whole group terminated.

We implemented background traffic using unicast flows in order to measure the traffic overhead incurred by multicast flows. There were 20 unicast flows in the network, each sending at a rate of 1 packet/s. We used a low unicast traffic load in order to observe the impact of the traffic overhead caused by the multicast flows rather than by the unicast flows themselves. The unicast senders and receivers were also randomly selected. The unicast routes were constructed by the Ad-hoc On-demand Distance Vector (AODV) protocol [11] built in the QualNet simulator.

For the experiments used to collect the average path lengths and numbers of forwarding nodes (Sections 3.1 and 3.2), we generated 10 different configurations. A configuration is determined by the node placement in the network and the set of multicast members. By changing the node placement and multicast membership, we obtained various configurations. For each configuration, we calculated the average path length and the number of forwarding nodes of the corresponding multicast tree. Each data point in the graphs discussed in Sections 3.1 and 3.2 is the average taken over the 10 configurations.

For the experiments related to the PDR, throughput, end-to-end delay and delay jitter metrics (Sections 3.3, 3.4 and 3.5), each data point in the graphs was obtained from one configuration and 10 runs using different randomly generated seed numbers. The collected data were then averaged over the 10 runs. In each experiments, we let the sources transmit data for 600 s of simulated time. After they stopped sending data, the simulation continued to run for 100 s of simulated time to give the last packets time to be processed and routed, for a total of 700 s.

To confirm the results reported in Sections 3.3, 3.4 and 3.5, we created two more configurations for each data point by changing the node placement in the network as well as unicast and multicast senders and receivers, and repeated the experiments. The results from these configurations are consistent with those presented in this paper.

All the graphs were plotted with a confidence interval of 95%.

The parameters common to all experiments are summarized in Table 1. Tables 2 and 3 listed the parameters specific to each set of experiments.

2.3. Assumptions

Our simulations are based on IEEE 802.11 CSMA/CA medium access control because this is a widely accepted radio technique for WMNs. Other types of WMNs that are being considered or standardized include 802.15 and 802.16 [1]. Since 802.16 uses Time Division Multiple Access (TDMA), and 802.15, a combination of TDMA and CSMA/CA, our future work is to extend this study to these types of networks.

We implemented only CSMA/CA without RTS/CTS for multicast medium access control, at both branch point and non branch point nodes in a multicast tree. There currently does not exist an effective algorithm for implementing RTS/CTS/DATA/ACK exchanges at the branch points of a multicast tree for the following two reasons. First, CTS packets sent by the multicast neighbors of a transmitter have a very high probability of colliding at the transmitter. More importantly, it may not be possible for all the

Table 1
Common simulation parameters

Parameter	Value
Network size	100 nodes over a 2000 m × 2000 m area 300 nodes over a 3000 m × 3000 m area
Path loss model	free space for distances below 250 m two-ray for distances of 250 m or longer
Router transmission power	20 dBm
Transmission rate at physical layer	11 Mbits/s
Physical layer protocol	PHY802.11b
Medium access control	MAC802.11 with DCF
MAC for multicast flows	CSMA/CA
MAC for unicast flows	CSMA/CA with RTS/CST/DATA/ACK
Packet size (excluding header size)	512 bytes
Queue size at routers	50 Kbytes
Queuing policy at routers	First-in-first-out
Traffic model of sources	Constant bit rate (CBR)
Confidence interval	95%

Table 2
More parameters for experiments discussed in Sections 3.1 and 3.2

Parameter	Values
Multicast group size	$\{10, 20, \ldots, 70, 80\}$ receivers in 100-node network $\{10, 20, 30, \ldots, 270, 280\}$ receivers in 300-node network
Number of configurations	10 per data point

Table 3
More parameters for experiments discussed in Sections 3.3, 3.4 and 3.5

Parameter	Values
Multicast group size	{20, 40, 80} receivers
Multicast sender's rate	$\{10, 20, 30, \ldots, 70, 80\}$ packets/s
Number of unicast flows	20
Each unicast sender's rate	1 packet/s
Unicast routing protocol	AODV
Duration of each experiment	700 s of simulated time
Number of runs per data point	10 per data point

multicast neighbors to agree on a common time slot for the transmission of a packet, or the delay would be very long to reach such an agreement. Therefore, all multicast implementations in 802.11-based wireless networks so far have used only CSMA/CA without RTS/CTS/DATA/ACK exchanges.

At non branch point nodes, we could implement RTS/CTS/DATA/ACK exchanges because the transmissions are in fact point-to-point. However, this would require more complex implementations to distinguish between branch point and non branch point nodes at the MAC layer, and to update node status (as branch point or not) when the tree is updated upon members joining or leaving. We thus favor the simpler implementation (i.e., without RTS/CTS).

We assume that all transmissions of a multicast group take place on one channel. Although multi-channel mesh networks have been studied extensively in order to enhance the overall network throughput, currently there are no effective multi-channel protocols for multicast communications, and designing such a protocol is not a trivial task either [13]. (Different multicast groups can use different channels though, as long as no channel switching is required for multicast transmissions. Our results are still valid when the multicast groups and unicast flows to be studied are on the same channel.)

We do not address specific protocols used by the wireless routers in order to build multicast trees efficiently in terms of communication and computation costs. Designing such protocols is outside the scope of this paper. We assume that the trees are given and we measure the performance of the trees after they are constructed. Nonetheless, the presented results can be used to guide the development of multicast routing protocols for WMNs (e.g., focusing on SPT protocols).

Finally, it is common knowledge that wireless multi-hop networks currently suffer from scalability issues; that is, when the network size increases, the network performance degrades considerably. Therefore, our simulations assume WMNs of small to medium sizes (compared with the Internet) such as community networking, intra- and inter-building enterprise networking, emergency ad-hoc networks, and metropolitan area networks [1].

3. Experimental results

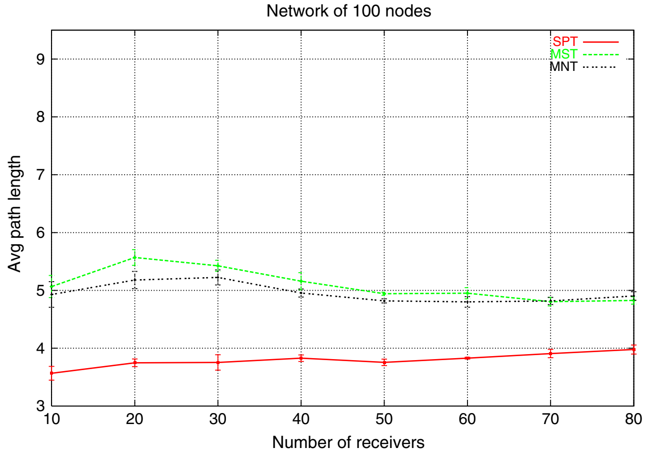
We compare the performance of multicast sessions created by the SPT, MST and MNT algorithms using the metrics listed in Section 2.1. We also examine the impacts of the traffic created by multicast nodes on other flows in the same network.

3.1. Average path lengths

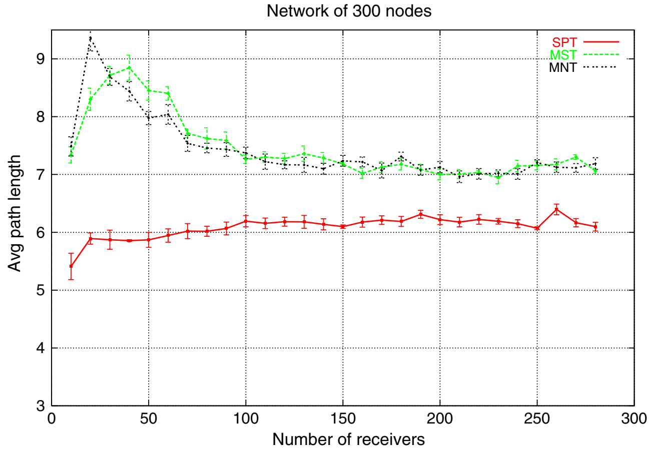
In this set of experiments, we simulated different multicast groups by varying the number of receivers from 10 to 80 in the small network, and 10 to 280 in the large network. Each multicast group has one source. The graph in Fig. 1 shows the average path lengths of the three types of trees. The results confirm that the MST and MNT algorithms produce longer paths than the SPT algorithm in all cases. Furthermore, the larger the network, the wider the difference gap. For instance, in the case of 40 receivers, the MST/MNT average path length is about 20% longer than the SPT average path length in the network of 100 nodes, but about 40% longer in the network of 300 nodes.

Let n be the group size (i.e., the number of receiver routers in the multicast group), N be the network size (i.e., the total number of routers in the WMN), and L_{SPT}, L_{MST} and L_{MNT} be the average path lengths of the SPT, MST and MNT, respectively. Following are our observations:

- Given the same multicast group, the SPT algorithm offers the lowest average path length, by definition of SPTs.
- In general, L_{SPT} tends to increase as n increases, since each path is built independently of the others as long as it is a shortest path from the source to the destination.



(a) Network of 100 nodes



(b) Network of 300 nodes

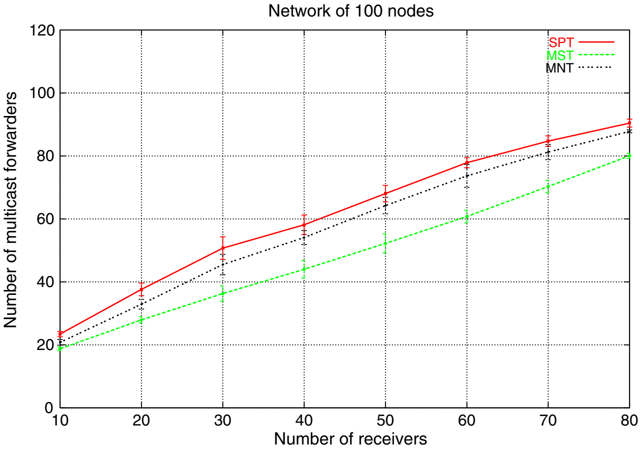
Fig. 1. Average path lengths.

- L_{MST} also increases as n increases for $n < 30$ in the 100-node network and $n < 50$ in the 300-node network. Nevertheless, after these points, L_{MST} decreases as n increases. At the beginning, because the multicast group is still small, a disjoint path (or sub-path) is usually created for each new receiver joining the group. The average path length thus increases. When the multicast group is large enough, the MST algorithm makes new receivers share as many links as possible with the existing receivers, because the algorithm tries to minimize the number of edges in the tree. Thus L_{MST} tends to decrease as more receivers join the multicast group. As n approaches N, the average path length does not change much, because the new receivers are nodes either already in the tree, or very close to the tree nodes.
- The MNT algorithm yields results similar to those of the MST algorithm with respect to the average path length metric. Moreover, the two algorithms give comparable average path lengths in most cases.

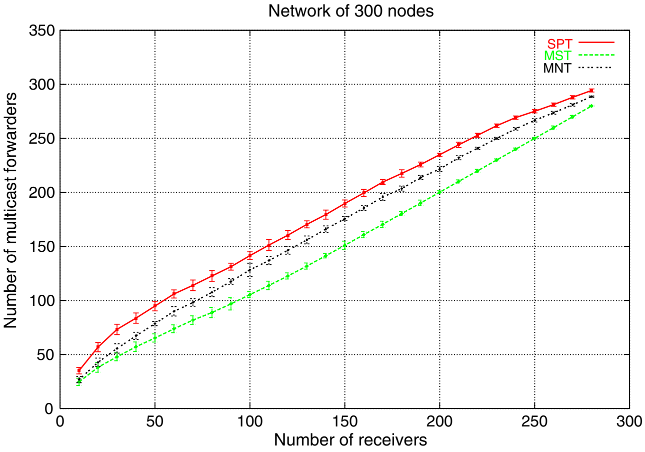
3.2. Numbers of forwarding nodes

In the above set of experiments (Section 3.1), we also recorded the number of forwarding nodes in each multicast tree. The results are shown in Fig. 2. Let F_{SPT}, F_{MST} and F_{MNT} be the number of forwarding nodes in the SPT, MST and MNT, respectively. We observe that

- F_{MST} goes up almost linearly as the group size n increases.
- F_{SPT} increases faster than F_{MST}, up to a certain point. An SPT path is established independently of the others in the current tree, while the MST algorithm makes new receivers share intermediate nodes with existing receivers to minimize the tree cost. This enables F_{SPT} to increase at a faster rate than F_{MST}. Above a certain group size ($n > 60$ in the small network, and $n > 220$ in the large network), the rate of increase of F_{SPT} slows down. As the group size approaches the network size, new paths are forced to share more intermediate nodes with existing paths, slowing down the the rate of increase of F_{SPT}.
- As n increases, all three values F_{SPT}, F_{MST} and F_{MNT} increases. Given the same multicast group, the SPT requires the most number of forwarding nodes as we would expect. On the other hand, it may seem surprising that the MNT heuristic by Ruiz and Gomez-Skarmeta did not create trees with a minimum number of forwarding nodes: in all cases, $F_{\mathrm{MST}} < F_{\mathrm{MNT}}$. In fact, the authors' results in [6] confirm ours: the proposed MNT heuristics are effective only in a network with high density (more than 40 nodes/km^2). In networks with low density such as those used in our experiments, the MST heuristic builds trees with both minimum total edge cost and a minimum number of transmissions.



(a) Network of 100 nodes (density = 25 nodes/km^2)



(b) Network of 300 nodes (density = 33 nodes/km^2)

Fig. 2. Numbers of multicast forwarding nodes.

Having analyzed the results with respect to the average path length and the number of forwarding nodes metrics, we may ask the question of how they affect the performance of a multicast group and other flows in terms of packet delivery ratio, throughput, end-to-end delay and delay jitter. In general, the longer the paths,

- the higher the probability a packet will be lost or damaged, hence the higher the loss rates of multicast receivers;
- the longer end-to-end delay;
- the more varied in delay (i.e., the higher the delay jitter).

The higher the number of forwarding nodes in a multicast tree, the more traffic it generates in the network, causing more congestion, channel contention and packet collisions. This negatively impacts the performance of both the multicast session and other flows in the network.

Compared with MSTs and MNTs, SPTs have the advantage of shorter path lengths, but the disadvantage of higher numbers of forwarding nodes. In the next sections, we explore how these two factors, path length versus number of forwarders, affect the other performance metrics.

3.3. Multicast performance in the small network

In the network of 100 nodes, we examined three multicast groups having 20, 40 and 80 receivers. The other parameters are summarized in Table 3. The results are illustrated in Figs. 3–5, respectively.

When the traffic load is light (under 30 packets/s), the performance of the three algorithms is comparable with respect to packet delivery ratio and throughput.

When the traffic load is moderate or high, the SPTs outperform the MSTs and MNTs in all cases, and the difference can be significant. For example, when the number of receivers is 20 and the traffic load is 60 packets/s, the PDRs of the SPT, the MST and the MNT are 97.2%, 83.5% and 84.8%, respectively (Fig. 3(a)). The throughput of the SPT is also the highest (Fig. 3(b)). This results from longer average path lengths of the MSTs and MNTs. The longer the path a packet has to travel, the higher its chance of getting damaged or lost due to collision and/or congestion, especially under high traffic loads.

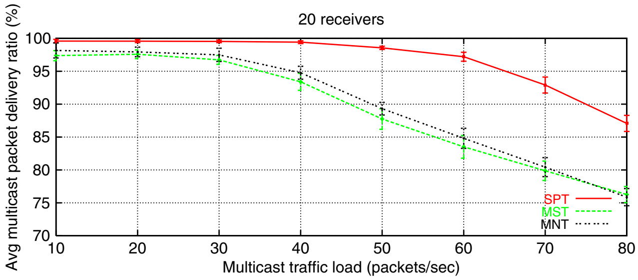
The average end-to-end delays incurred by the SPTs are the lowest in almost all cases, thanks to shorter source-to-destination paths. For instance, in the case of 20 receivers and 40 packets/s, the average end-to-end delays given by the SPT and the MST are 10.5 ms and 18.2 ms, respectively (Fig. 3(c)); in other words, the MST average end-to-end delay is almost 73% higher. The MNT provides the highest average end-to-end delay in this case, 18.8 ms.

The SPTs also give the lowest average delay jitter values in almost all the cases as a result of shorter average path lengths.

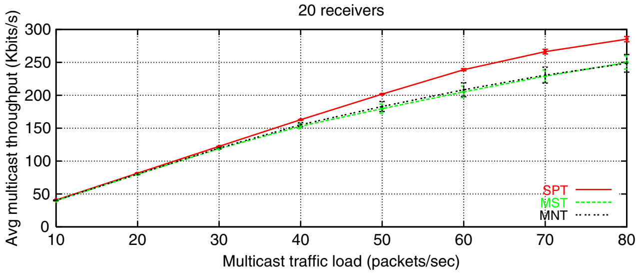
While SPTs give better performance than MSTs in almost all the cases, the performance gap narrows down as the group size n increases; compare the graphs in Figs. 3(a), 4(a) and 5(a) for an example. The reason is that when n increases from 20 to 40 and then 80, L_{SPT} increases while L_{MST} decreases. Moreover, F_{SPT} increases more quickly than F_{MST}. Since an increase in L or F value has a negative impact on the multicast tree, the MST/MNT performance approaches the SPT performance as the group size enlarges.

In fact, as n approaches the network size, the three trees, SPT, MST and MNT, have a large number of intermediate nodes in common, making them almost the same. Therefore, the performance of the three trees is similar, especially with respect to the PDR and throughput, as illustrated by the graphs in Fig. 5 where $n = 80$ receivers.

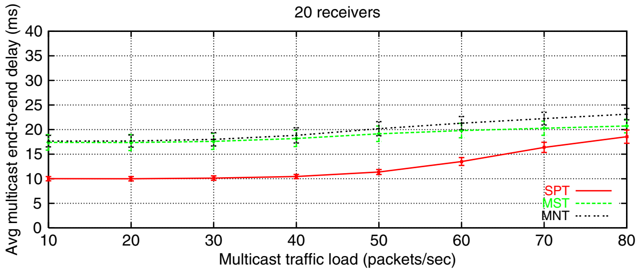
We now examine the performance changes of SPTs as a function of the group size n. The graphs in Fig. 6, which are compiled from the SPT data in Figs. 3–5, show that when the traffic load is light to moderate (under 50 packets/s in this set of experiments), groups of different sizes give similar performance in terms of PDR (Fig. 6(a)) and end-to-end delay (Fig. 6(b)). Although the number of forwarding nodes increases as the group size enlarges (Fig. 2(a)), it does not affect the multicast performance much under light traffic. However, at higher loads (50



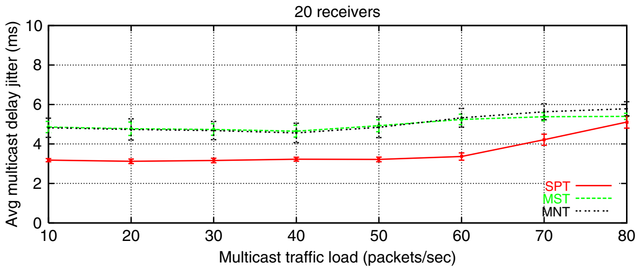
(a) Multicast PDR



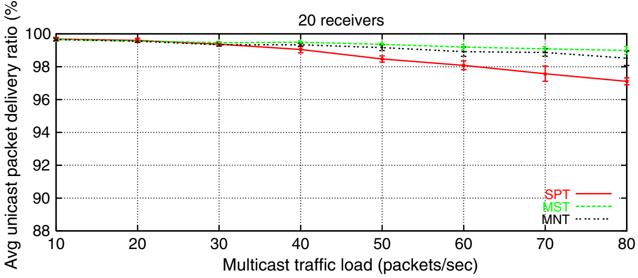
(b) Throughput



(c) End-to-end delay

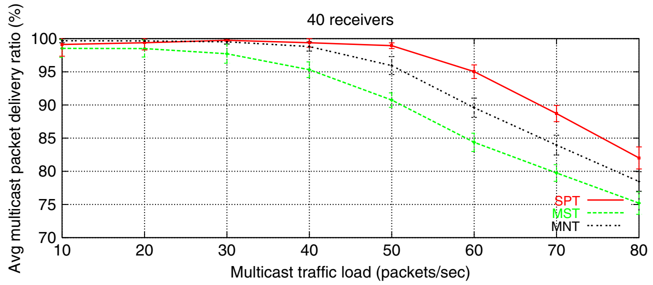


(d) Delay jitter

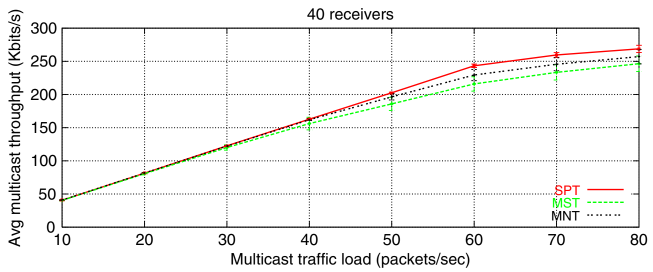


(e) Unicast PDR

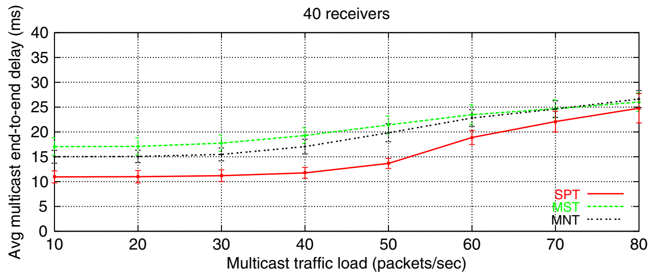
Fig. 3. SPT versus MST: small network of 100 nodes, 20 receivers.



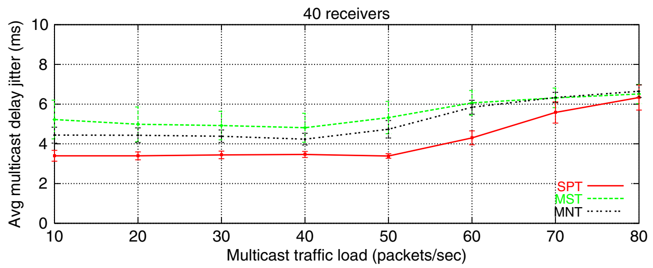
(a) Multicast PDR



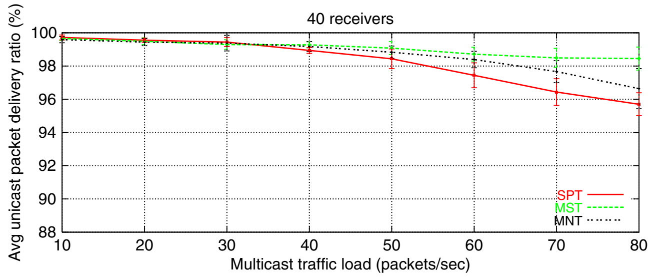
(b) Throughput



(c) End-to-end delay

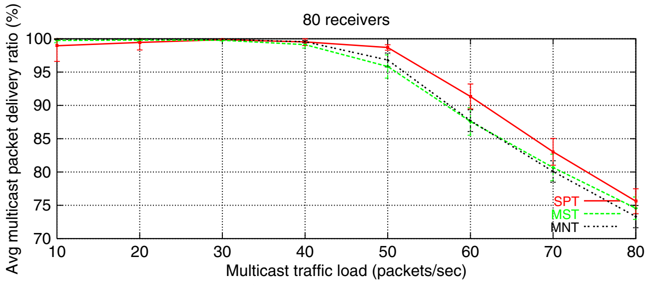


(d) Delay jitter

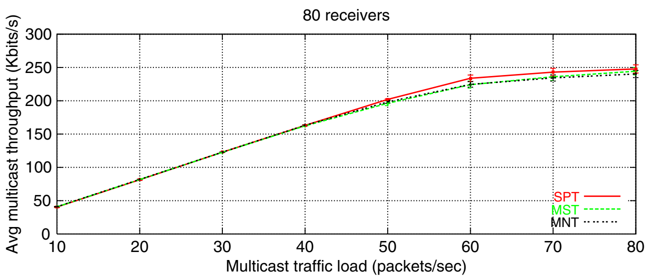


(e) Unicast PDR

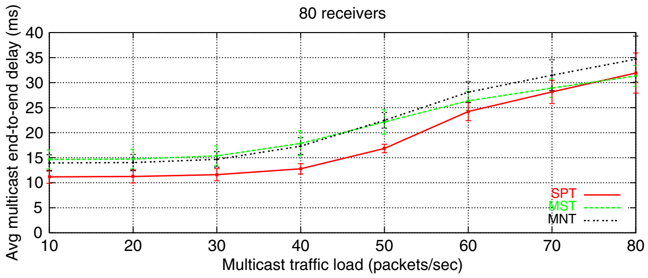
Fig. 4. SPT versus MST: small network of 100 nodes, 40 receivers.



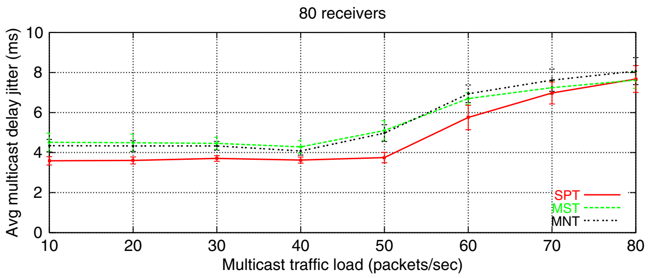
(a) Multicast PDR



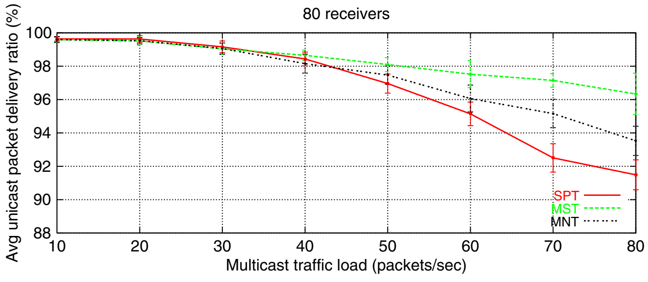
(b) Throughput



(c) End-to-end delay



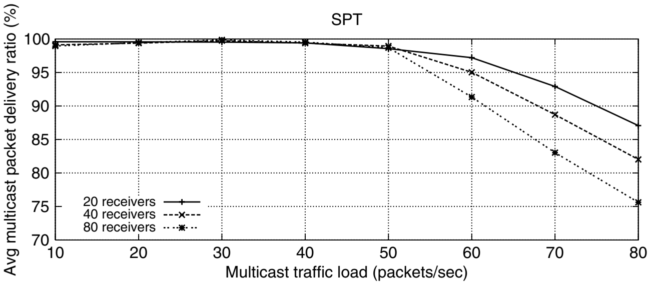
(d) Delay jitter



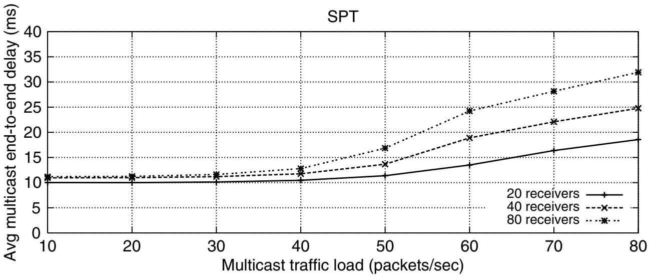
(e) Unicast PDR

Fig. 5. SPT versus MST: small network of 100 nodes, 80 receivers.

packets/s or more), large groups perform much worse than small groups, as a result of the increase in the number of forwarding nodes compounded with high traffic loads. The same observations are applicable to the throughput and delay jitter metrics, which can be inferred from Figs. 3–5.



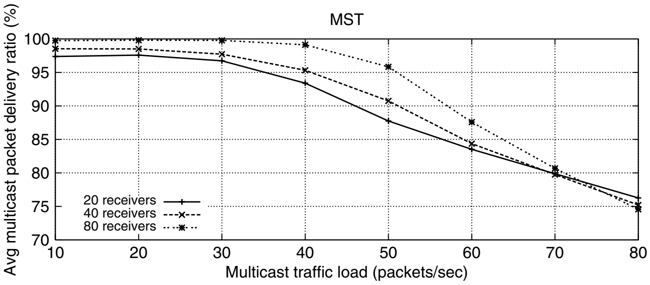
(a) Multicast PDR



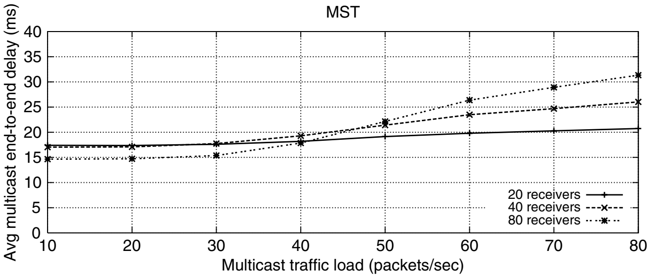
(b) End-to-end delay

Fig. 6. SPT performance: network of 100 nodes.

When the sender's rate is low or moderate, MSTs perform somewhat differently from SPTs, as illustrated by the graphs in Fig. 7, which are also compiled from Figs. 3–5. Large MST groups perform better than small groups. As mentioned above, the impact of the number of forwarding nodes does not show under light loads. On the other hand, the average path length of MSTs goes down as the group size enlarges (Section 3.1) and this helps improve the performance. Yet, when the traffic load is high, the performance of MSTs degrades quickly as the group size increases (similarly to the SPT case). For example, the gap between the 80-receiver and 20-receiver PDR curves in Fig. 7(a) narrows down as the sending rate goes from 50 packets/s to 60 packets/s. This results from a combination of heavy traffic and the increase in the number of forwarding nodes, as with SPTs.



(a) Multicast PDR



(b) End-to-end delay

Fig. 7. MST performance: network of 100 nodes.

The above observations and explanations for MSTs also apply to MNTs. In fact, the two algorithms give similar multicast performance in most of the cases.

3.4. Multicast performance in the large network

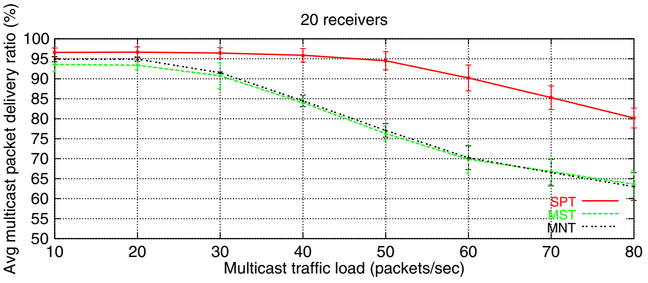
Figs. 8–10 show the results of three multicast groups having 20, 40 and 80 receivers, respectively, in the large network of 300 nodes. As in the smaller network, SPTs perform better than MSTs and MNTs in terms of PDR and throughput, and in almost all cases, end-to-end delay and delay jitter. The performance differences between SPTs and MSTs/MNTs are even more pronounced in the large network. For instance, when the number of receivers is 40 and the traffic load is 60 packets/s, the PDRs of the SPT and the MST in the network of 100 nodes are 95% and 84.3%, respectively (Fig. 4(a)), while the PDRs in the network of 300 nodes are 93% and 70%, respectively (Fig. 9(a)). In the same scenario, the average end-to-end delay given by the MST is about 24% higher than that of the SPT in the smaller network (23.5 ms versus 18.9 ms in Fig. 4(c)), and about 54% higher in the larger network (39.8 ms versus 25.9 ms in Fig. 9(c)). In other words, given the same multicast group size, as the network size increases, the performance gap between SPTs and MSTs/MNTs widens. The reason is that the larger the network, the bigger the difference in path length between the SPT and the MST/MNT as shown in Section 3.1.

In the case of the 80-receiver group (Fig. 10), it may be noted that under high traffic loads of 70 or 80 packets/s, SPTs incur higher end-to-end delay and/or delay jitter than MSTs/MNTs. This is the result of SPTs using more forwarding nodes than MSTs/MNTs, (e.g., 27% more than MST when $n = 80$ from Fig. 2(b)), which causes more network congestion and channel contention. Nonetheless, the PDRs of MSTs and MNTs drop to under 72% in these cases, which is not acceptable in most real-life applications. Thus, in practice, SPTs should be used and the sending rate should be limited to under 60 packets/s to lower the amount of multicast traffic in the network.

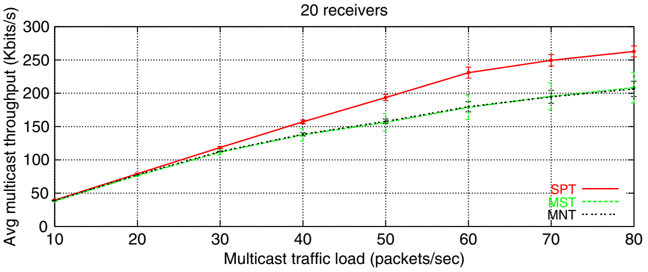
Again, the multicast performance of MSTs and MNTs is comparable in most cases.

3.5. Impacts of multicast traffic on other flows

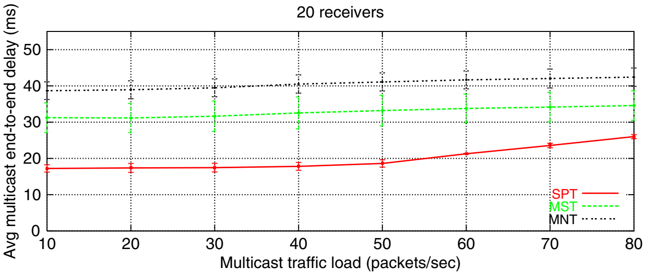
In the small network of 100 nodes, the three algorithm cause similar loss rates to the unicast flows. Only when the group size is large and the sending rate is high does SPT cause more packet loss than the other two algorithms, up to 3% more in practical cases (i.e., when the sending rate



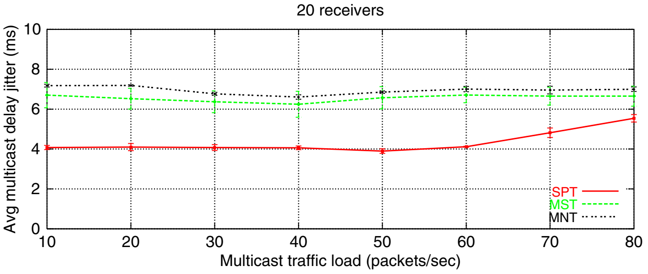
(a) Multicast PDR



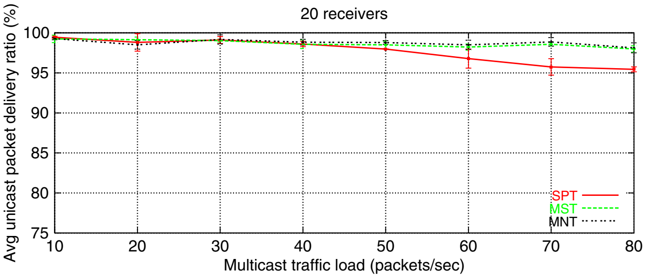
(b) Throughput



(c) End-to-end delay

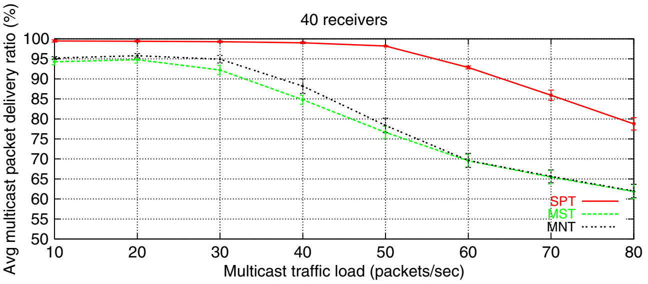


(d) Delay jitter

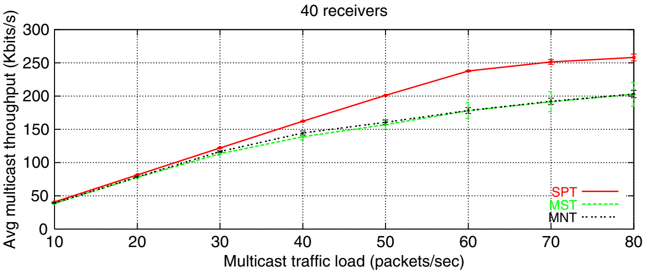


(e) Unicast PDR

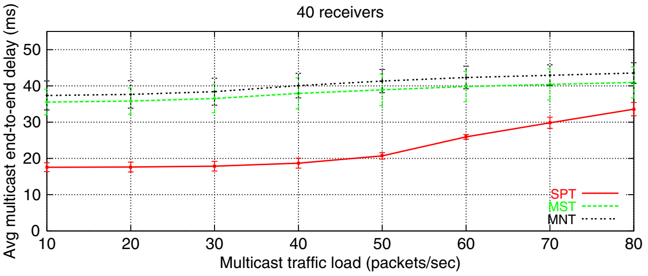
Fig. 8. SPT versus MST: large network of 300 nodes, 20 receivers.



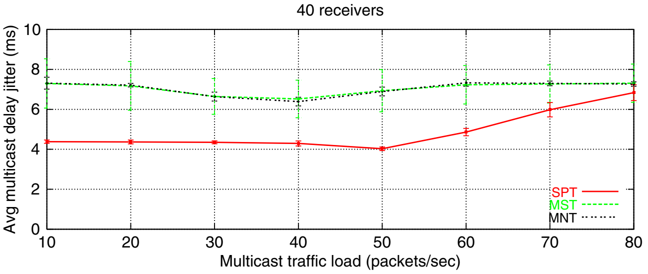
(a) Multicast PDR



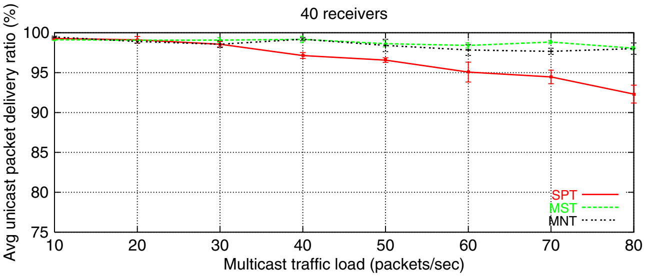
(b) Throughput



(c) End-to-end delay



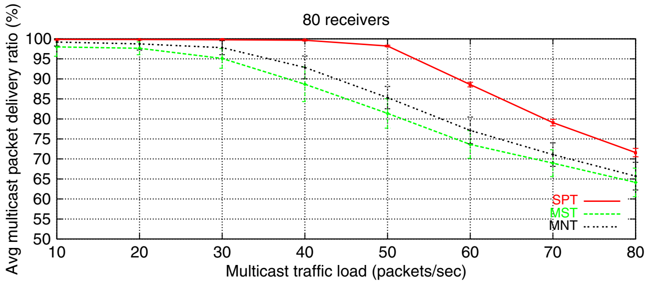
(d) Delay jitter



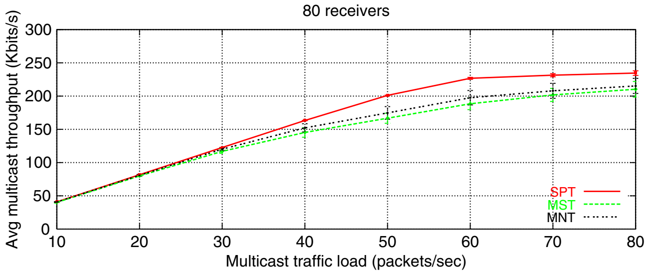
(e) Unicast PDR

Fig. 9. SPT versus MST: large network of 300 nodes, 40 receivers.

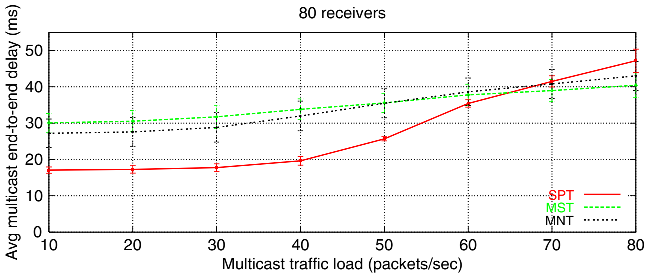
is 60 packets/s or less in order to obtain acceptable PDRs). This results from higher numbers of forwarding nodes in the SPTs. If we weigh the multicast performance gain against the unicast PDRs in the case of 20 and 40 receivers (Figs. 3 and 4), the SPT algorithm is the better choice. In the case of 80 receivers (Figs. 5), MSTs/MNTs may be



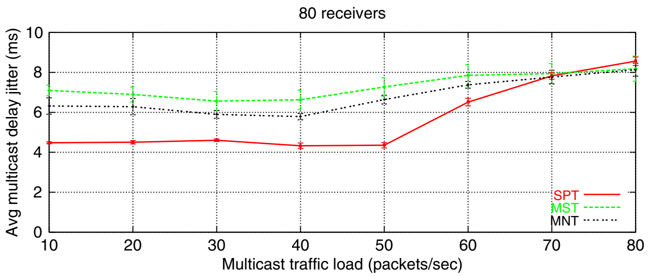
(a) Multicast PDR



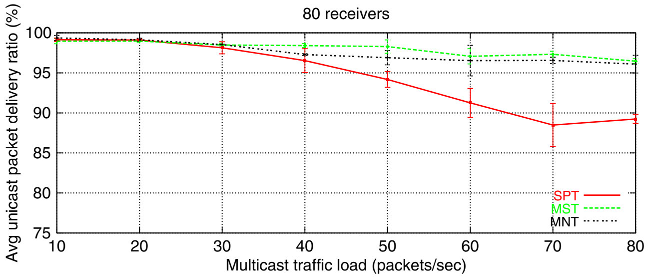
(b) Throughput



(c) End-to-end delay



(d) Delay jitter



(e) Unicast PDR

Fig. 10. SPT versus MST: large network of 300 nodes, 80 receivers.

preferable because their multicast performance is not far from that of SPTs, yet they cause less losses to the unicast flows.

In the 300-node network, we would also choose SPTs because of the multicast performance gain versus traffic overhead as explained above. For instance, when the group size is 80 receivers and the load is 50 packets/s, the SPT causes about 3% higher unicast loss rate than the MNT, but gives 14% higher multicast PDR (Fig. 10). Note that for this group size we should limit the source rate to 50 packets/s or lower to obtain acceptable multicast delivery ratios. In this case, the SPT algorithm offers considerably higher multicast performance than the MST and MNT while incurring an acceptable amount of traffic overhead. Overall, in the 300-node network, SPTs cause up to 5% more packet loss than MSTs/MNTs in practical cases.

As discussed in Section 3.3, given the same network and parameters, as the group size increases, the multicast performance gap between SPTs and MSTs/MNTs narrows down. In contrast, the unicast loss rate (the traffic overhead) gap widens as the number of receivers increases; compare Figs. 8(e), 9(e) and 10(e) for an example. The reason is that as more members join a group, F_{SPT} increases at a faster rate than F_{MST} and F_{MNT} (Fig. 2(b)), widening the difference gap with respect to traffic overhead.

3.6. Summary of experimental results

Among the three algorithms we examined, the MST heuristic creates trees with both minimum total edge cost and minimum number of transmissions. The SPT algorithm gives shortest average path lengths.

In our experiments, the MNT heuristic proposed by Ruiz and Gomez-Skarmeta did not build optimal trees in terms of number of transmissions. In fact, these results are consistent with the finding from the paper by Ruiz and Gomez-Skarmeta: the proposed MNT heuristics work effectively only in high density networks (about 40 nodes/km^2 or more [6]). WMNs are in general of low density (less than 40 nodes/km^2) compared with other types of wireless multi-hop networks such as mobile ad-hoc and sensor networks, because transmission ranges of mesh routers are typically much longer (more than 300 m [10,17,18]).

When the multicast group size is small to medium (relative to the network size, e.g., $n \leqslant 50$ in the 100-node network, and $n \leqslant 100$ in the 300-node network), SPTs offer significantly better performance to multicast flows than MSTs and MNTs. This results from shorter path lengths in SPTs, which gives higher delivery ratio and throughput, as well as lower end-to-end delay and delay jitter.

For large groups with low multicast rates, SPTs also outperform MSTs and MNTs, although to a lesser extent, and the three types of trees incur similar multicast traffic overhead.

However, when the multicast group size is large and the source rate is high (60 packet/s or more in our experiments), SPTs cause more packet loss to other flows in the network than MSTs and MNTs, due to higher numbers of forwarding nodes in SPTs combined with heavy traffic loads. If we consider only the practical cases, SPTs caused up to 3% more unicast packet loss in our simulated 100-node network, and up to 5% in the 300-node network. Nonetheless, in those cases, SPTs yield *considerably* higher PDRs and lower end-to-end delay and delay jitter.

Based on the experimental results, we offer the following recommendations.

4. Recommendations and discussions

We would recommend the SPT algorithm for groups having small to medium sizes (relative to the network size), or large groups with low sending rates (e.g., source rate of 50 packets/s or less), because SPTs perform better than MSTs and MNTs, and the three types of trees incur comparable traffic overhead.

When the group size is large and the sending rate is high, large numbers of forwarding nodes in SPTs may become a concern. However, if we weigh the multicast performance gain against the traffic overhead of SPTs, these trees are still recommended

Another reason for the recommendation stated in the above paragraph is that it is much easier to design a reliable transport protocol for unicast communications than for multicast communications in wireless multi-hop networks. Several TCP-based protocols have been proposed for reliable data delivery for unicast flows in wireless ad-hoc networks [14], which could be applied to WMNs. However, the problem of reliable multicast in wireless ad-hoc networks still remains under-researched. Although several reliable multicast protocols have been proposed for the Internet (e.g., SRM, RMP, RMTP [2], NORM and ALC [15]), their applicability to and efficiency in WMNs have not been studied. Until a reliable multicast protocol is proved effective and efficient for use in WMNs, SPTs provide the best trade-off in terms of multicast and unicast packet delivery ratios. The unicast packets lost can be retransmitted and delivered using the existing TCP-based protocols [14].

Even if an efficient reliable multicast protocol were available for use in WMNs, SPTs would still be considered the better choice for many real-time applications such as video/audio conferencing, video streaming, distance learning, multi-party interactive games, and distribution of time-sensitive data (e.g., stock quotes, news), because SPTs offer lower end-to-end delay and delay jitter than MSTs and MNTs.

To alleviate the impacts of SPT multicast traffic on the network, we can limit the multicast sending rate by effective flow control mechanisms. For applications requiring high rates (e.g., videos), a dedicated channel can be used for multicast in a multi-channel network. Multi-channel and multi-radio systems have been proved to be the technology for high-performance WMNs and implemented by several wireless mesh product manufacturers [17,18].

We should also consider another advantage of SPTs over MCTs such as MSTs and MNTs. Although the wireless mesh routers are static, mesh hosts attached to the wireless end-routers such as cell phones, PDAs, and laptops may move from one wireless end-router to another, and may join and leave a multicast session freely at will. It is much easier to support dynamic joins and leaves using SPTs than MCTs, because in a SPT each source-to-destination path is established independently of the other paths in the tree. In a MCT, a node joining or leaving the multicast session may require the whole tree to be re-computed in order to maintain the cost optimality (or the new tree would no longer be optimal). This implementation difficulty of MCTs makes SPTs the more practical multicast routing approach in any kind of network.

5. Related work

In this section, we review existing work on multicast routing in different kinds of networks: wireline Internet, mobile ad-hoc networks (MANETs) and WMNs. For each approach or protocol, we briefly analyze its applicability and efficiency for use in WMNs.

5.1. Multicast routing on the Internet

The majority of the multicast routing protocols used in the Internet today are based on shortest path trees because they are easy to implement and they provide minimum delay from the sender to each receiver, which is a desirable property for most real-life multicast applications. For example, MOSPF [19] uses the Dijkstra's shortest path algorithm and DVMRP [20] is a distributed implementation of the Bellman–Ford shortest path algorithm. Although these protocols work well in wired networks, they are not suitable for wireless environments. DVMRP requires a multicast sender to periodically flood the network in order to prune or graft branches to keep the multicast tree up-to-date. In MOSPF, when a member joins or leaves the group, a membership update message is flooded to the entire MOSPF routing domain to inform the routers in the domain. Flooding is very expensive in a wireless environment because it consumes bandwidth and causes channel contention and packet collisions [12]. A more efficient mechanism for handling membership updates is needed for constructing multicast trees in WMNs.

There also exist algorithms that try to optimize on both overall cost and source-to-destination distance (or end-to-end delay) [21,22]. These algorithms, however, are not commonly used due to complex implementations.

More recently we have seen a few multicast routing protocols in the Internet that are based on sub-optimal shared trees as opposed to per-source SPTs. Examples of those protocols include Protocol Independent Multicast (PIM) [23] and Core-Based Tree (CBT) [24]. PIM introduces the notion of a rendez-vous point (RP), which acts as a meeting place of the receivers and the senders. The receivers explicitly join a tree (called shared RP-tree) rooted at the RP. A source will then send data to the RP which will then relay (by multicasting) the data to the receivers in the shared RP-tree. In a many-to-many multicast group with *m* senders, MOSPF or DVMRP would require *m* trees, while PIM needs to create and maintain only one shared tree. PIM thus helps minimize the amount of routing information to be stored by the forwarding routers, especially when the multicast group size and the number of senders are large. Similarly, CBT sets up a single shared bidirectional tree connecting the senders and receivers. All senders use the same tree for transmitting the data. The routers simply forward the data on all the interfaces for the CBT except for the incoming interface. It should be emphasized that the reason for using PIM or CBT shared trees (as opposed to per-source trees) is to reduce the amount of routing information to be stored in the routers rather than to minimize the overall cost. Compared with per-source SPTs, shared-tree algorithms produce source-to-destination paths longer than necessary. Longer paths potentially result in higher loss rates and longer end-to-end delay. Future work is needed to quantify the performance differences between shared trees and per-source trees.

5.2. Multicast routing in mobile ad-hoc networks

Early efforts in multicast routing in MANETs were to adapt traditional tree-based algorithms to MANETs, resulting in a class of protocols including MAODV [31], AMRoute [32], and AMRIS [33]. Tree-based protocols do not perform well in MANETs, nonetheless, because of node mobility. When nodes move, the single path between a source and a receiver may break, resulting in high loss rates. As a result, a new category of protocols was designed that builds routing meshes rather than routing trees. In a routing mesh, there exist several paths between a source and a receiver. If a path breaks, duplicate copies of a packet may still be able to reach the destinations via alternate paths, resulting in better packet delivery ratios. Experimental results have shown that mesh-based protocols outperform tree-based protocols with respect to packet delivery ratio [34]. Typical mesh-based protocols are ODMRP [34] and CAMP [35].

When applied to WMNs, routing meshes may offer better PDR than routing trees under very light traffic load due to path redundancy [12]. However, as the group size and traffic load increase, their performance becomes worse than that of routing trees because more nodes are involved in the forwarding mesh, resulting a high number of transmissions. This also creates more traffic in the network and negatively affects other flows [12].

5.3. Multicast routing in wireless mesh networks

Ruiz and Gomez-Skarmeta redefine the cost of minimum cost trees for wireless multi-hop networks [6,25]. In this case, a minimum cost tree is one that contains a minimum number of forwarding nodes, and thus issuing a minimum number of transmissions. The authors proposed two heuristics to compute such trees. Our experimental results presented in this paper show that the proposed heuristics are not effective in networks with low density such as wireless mesh backbones. As a matter of fact, the trees created by the MNT heuristic always contain more forwarding nodes than those by the MST algorithm in our experiments.

Zhao et al. extend the SPT and MNT algorithms to provide a pair of paths between the sender and each receiver for more reliable delivery [36]. Although double-path routing may improve the multicast delivery ratio compared with single-path routing, it also creates more traffic, and potentially causes higher loss rates to other flows in the network. The paper did not provide experimental results with respect to PDR, throughput or end-to-end delay. Future work is needed to justify the trade-off between the performance gain of double-path routing (if any) and the resulting traffic overhead.

Roy et al. [37] adapt five routing metrics designed for unicast routing, namely Expected Transmission Count (ETX) [27], Expected Transmission Time (ETT) [26], Packet Pair (PP) [28], Multicast ETX (METX) [29], and Success Probability Product (SPP) [30], to multicast routing. They studied the performance of these metrics on the multicast routing protocol ODMRP [34] in WMNs. The authors' experimental results show that SPP and PP offer the best performance, achieving 14% to 18% higher throughput than the original ODMRP. The above metrics require nodes to periodically broadcast probe packets to their neighbors in order to measure link quality and bandwidth. Estimating link quality and bandwidth by probes is currently unreliable [38–40] and incurs high traffic overhead. More effective and efficient methods are needed to improve path metric computation and route selection before they can be applied to multicast routing in WMNs.

Yuan et al. [41] propose a cross-layer optimization framework that balances the supply of link capacities at the physical layer and the demand of network flows at the network layer in order to find high throughput paths. The authors combined the network coding technique [42,43] for multicast routing and a game theoretic method [44] for interference management at the physical layer. The complexity and feasibility of implementing this protocol in real networks are to be determined.

Research on multicast in WMNs is still in its infancy. Open issues include reliable multicast, flow/congestion control, efficient membership updates, routing in multi-channel, multi-radio networks, quality-of-service guarantees, and security provisioning (e.g., authentication, access control, and group key management).

6. Conclusions and future work

We quantify the performance differences of minimum cost trees (MCTs) and shortest path trees (SPTs) in WMNs. For MCTs, we consider both minimum Steiner trees (MSTs) and minimum number of transmissions trees (MNTs). Our simulation results show that SPTs offer significantly better performance to multicast flows than MCTs in terms of PDR, throughput, end-to-end delay and delay jitter. The only drawback of SPTs is that when the group size is large and the multicast sending rate is high SPTs cause more packet losses to other flows than MCTs. By weighing the high multicast performance gain against the impact of traffic overhead of SPTs, we would recommend the SPT approach for multicast routing in WMNs. To lessen the impacts of SPT multicast traffic on the network, we can limit the multicast sending rate via flow control and take advantage of multi-channel, multi-radio systems. Other advantages of SPTs over MCTs include lower end-to-end delay and delay jitter, which are important metrics in real-time and audio/video applications, and much easier support for dynamic joins and leaves.

The work presented in this paper is the first step towards a large project that aims at providing a suite of protocols for efficient group communications in WMNs. Our future work includes efficient membership updates in WMNs, many-to-many routing, quality-of-service routing, reliable multicast and flow/congestion control for multicast. We will also study these issues in 802.15 and 802.16 networks.

Acknowledgements

This work is supported in part by a Discovery Grant from the Natural Sciences and Engineering Research Council of Canada (NSERC). We would like to thank our students Jin Xu, Marvin Pinto and Yi Zheng for running the experiments, collecting the data and plotting the graphs, and Celia Li and Lan Nguyen for proofreading the paper.

References

[1] I. Akyildiz, X. Wang, W. Wang, Wireless mesh networks: A survey, Computer Networks 47 (4) (2005) 445–487.
[2] S. Paul, Multicast on the Internet and Its Applications, Kluwer Academic Publishers, 1998.
[3] D. Bertsekas, R. Gallagher, Data Networks, Prentice Hall, 1991.
[4] L. Kou, G. Markowsky, L. Berman, A fast algorithm for Steiner trees, Acta Informatica 2 (15) (1981) 141–145.
[5] A. Zelikovsky, An 11/6-approximation algorithm for the network Steiner problem, Algorithmica 9 (5) (1993) 463–470.
[6] P.M. Ruiz, A.F. Gomez-Skarmeta, Approximating optimal multicast trees in wireless multihop networks, in: 10th IEEE Symposium on Computers and Communications, June 2005, pp. 686–691.
[7] QualNet Network Simulator version 3.9.1, <http://www.scalable-networks.com/>.
[8] GloMoSim Simulator, <http://pcl.cs.ucla.edu/projects/glomosim/>.
[9] S. Saunders, Antennas and Propagation for Wireless Communication Systems, Wiley, 2000.
[10] TROPOS Networks, <http://www.tropos.com>.
[11] C.E. Perkins, E.M. Royer, S.R. Das, Ad hoc on demand distance vector (AODV) routing, in: IEEE WMCSA, 1999.
[12] J. Xu, Multicast in Wireless Mesh Networks, Master's Thesis, Department of Computer Science and Engineering, York University, Toronto, Canada, October 2006.
[13] P. Kyasanur, J. So, C. Chereddi, N.H. Vaidya, Multi-channel mesh networks: challenges and protocols, IEEE Wireless Communications 13 (2) (2006).
[14] C. Siva Ram Murthy, B.S. Manoj, Ad hoc Wireless Networks: Architectures and Protocols, Prentice Hall, 2004.
[15] Reliable Multicast Transport, Internet Engineering Task Force, <http://www.ietf.org/html.charters/rmt-charter.html>.
[16] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms, The MIT Press, 2001.
[17] Mesh Dynamics, <http://www.meshdynamics.com>.
[18] Bel Air, <http://www.belair.com>.
[19] J. Moy, Multicast extensions to OSPF, in: IETF RFC 1584, 1994.
[20] D. Waitzman, C. Partridge, S. Deering, Distance vector multicast routing protocol, in: IETF RFC 1075, 1988.
[21] B. Kadaba, J.M. Jaffe, Routing to multiple destinations in computer networks, IEEE Transactions on Communications 31 (3) (1983) 343–351.
[22] V.P. Kompella, J.C. Pasquale, G.C. Polyzos, Multicast routing for multimedia communication, IEEE/ACM Transactions on Networking 1 (3) (1993) 286–292.
[23] S. Deering, D. Estrin, D. Farinacci, V. Jacobson, C. Liu, L. Wei, The PIM architecture for wide-area multicast routing, IEEE/ACM Transactions on Networking 4 (2) (1996).
[24] T. Ballardie, P. Francis, J. Crowcroft, Core bared trees (CBT): An architecture for scalable multicast routing, in: ACM SIGCOMM, 1995, pp. 85–95.
[25] P.M. Ruiz, F.J. Galera, C. Jelger, T. Noel, Efficient multicast routing in wireless mesh networks connected to Internet, in: 1st International Conference on Integrated Internet Ad hoc and Sensor Networks, May 2006.
[26] R. Draves, J. Padhye, B. Zill, Routing in multi-radio, multi-hop wireless mesh networks, in: ACM MobiCom, September 2004.
[27] D. Couto, D. Aguayo, J.C. Bicket, R. Morris, A high-throughput path metric for multi-hop wireless routing, in: ACM MobiCom, 2003.
[28] S. Keshav, A control-theoretic approach to flow control, in: ACM Conference on Communications Architecture and Protocols, 1991, pp.3–15.
[29] Q. Dong, S. Banerjee, M. Adler, A. Misra, Minimum energy reliable paths using unreliable wireless links, in: ACM MobiHoc, 2005.
[30] S. Banerjee, A. Misra, Minimum energy paths for reliable communication in multi-hop wireless networks, in: ACM MobiHoc, 2002.
[31] E.M. Royer, C.E. Perkins, Multicast operation of the ad hoc on-demand distance vector routing protocol, in: ACM MobiCom, 1999.
[32] Jason Xie, Rajesh R. Talpade, Anthony McAuley, Mingyan Liu, AMRoute: ad hoc multicast routing protocol, ACM/Kluwer Mobile Networks and Applications 7 (6) (2002).
[33] C.W. Wu, Y.C. Tay, C.K. Toh, Ad hoc multicast routing protocol utilizing increasing id-numberS (AMRIS) functional specification, in: IEEE MILCOM, 1999, pp. 25–29.
[34] S.J. Lee, W. Su, M. Gerla, On-demand multicast routing protocol in multihop wireless mobile networks, ACM/Kluwer Mobile Networks and Applications 7 (6) (2002).
[35] J.J. Garcia-Luna-Aceves, E.L. Madruga, The core-assisted mesh protocol, IEEE Journal on Selected Areas in Communications 17 (8) (1999) 1380–1994.
[36] X. Zhao, C.T. Chou, J. Guo, S. Jha, Protecting multicast sessions in wireless mesh networks, in: 31st IEEE Conference on Local Computer Networks, November 2006, pp. 467–474.
[37] S. Roy, D. Koutsonikolas, S. Das, Y.C. Hu, High-throughput multicast routing metrics in wireless mesh networks, in: 26th International Conference on Distributed Computing Systems, July 2006.

[38] A. Adya et al., A multi-radio unification protocol for IEEE 802.11 wireless networks, in: IEEE First Conference on Broadband Networks, 2004, pp. 344–354.

[39] R. Draves, J. Padhye, B. Zill, Routing in multi-radio, multi-hop wireless mesh networks, in: ACM MobiCom, 2004.

[40] R. Draves, J. Padhye, B. Zill, Comparison of routing metrics for static multi-hop wireless networks, in: ACM SIGCOMM, 2004.

[41] J. Yuan, Z. Li, W. Yu, B. Li, A cross-layer optimization framework for multihop multicast in wireless mesh networks, IEEE Journal on Selected Areas in Communications 24 (11) (2006).

[42] R. Ahlswede, N. Cai, S.R. Li, R.W. Yeung, Network information flow, IEEE Transactions in Information Theory 46 (4) (2000) 1204–1216.

[43] Z. Li, B. Li, D. Jiang, L.C. Lau, On achieving optimal throughput with network coding, in: IEEE INFOCOM, March 2005.

[44] J.M. Osborne, An Introduction to Game Theory, Oxford University Press, 2004.