A Multichannel TDMA MAC Protocol to Reduce End-to-End Delay in Wireless Mesh Networks

Tran Minh Trung and Jeonghoon Mo

Supporting QoS over multihop wireless mesh networks is difficult because end-to-end delay increases quickly with the increasing number of hops. This paper introduces a novel multichannel time-division multiple-access media access control (McTMAC) protocol that can help to efficiently reduce delay over multihop networks. Performance evaluation results demonstrate that McTMAC outperforms existing alternative protocols. The max-delay can be reduced by as much as 60% by using McTMAC.

Keywords: Multichannel, TDMA, scheduling delay, time-slot assignment, channel assignment.

I. Introduction

A wireless mesh network (WMN) is an emerging communication network which consists of radio nodes in a mesh topology. With its popularity, supporting quality of service (QoS) over multihop radio links is becoming an issue because end-to-end delay increases quickly with the increasing number of hops. To solve the delay problem, a time-division multiple-access (TDMA)-based media access control (MAC) was incorporated into the standard [1]. In [2], a time-slot allocation algorithm was proposed for TDMA-based WMNs to minimize end-to-end delay. An optimization problem was formulated to minimize the maximum delay; however, it is limited in that it assumes a single-channel system.

Though there are many channel allocation algorithms for mesh networks [3]-[5], most of them are targeted toward capacity enhancements. Aryafar and others [3] proposed a channel allocation algorithm called "distance-1" to increase system capacity. Although the algorithm can significantly improve system capacity, it does not consider end-to-end delay. Yu and others [4] proposed algorithms to find suboptimal channel assignment and a centralize link schedule for multi-radio multichannel WMN based on a max-flow graph. These algorithms can minimize the number of time slots required to transport all the data. However, minimizing the number of time slots cannot guarantee that the end-to-end delay of a multihop flow is reduced [2]. Yun and others [5] proposed centralized and distributed maximal scheduling algorithms for multi-radio multichannel WMN. They took into account the switching delay, but they did not consider the transmission order of links in multihop connections, which is an important factor that effects end-to-end delay [2].

In this letter, we extend the algorithm proposed in [2] to the multichannel environment. More specifically, we consider a multichannel TDMA-based mesh network and propose channel and time-slot allocation algorithms with an objective of reducing delay. Our work is different from previous studies in that our algorithms are based on the length of flow to perform channel assignment and time-slot allocation.

II. Problem Statement

Consider a multihop WMN, $G=(N, L)$, that consists of a set of devices, N, and a set of links, L, between devices. A link $l=(n, m)$ exists between nodes n and m if n and m are within transmission range. If $l \in L$, n sends a packet to m using one of C channels as the system has multiple channels. Moreover, we assume a TDMA-type system, in which the time is divided into

Manuscript received Mar. 29, 2010; revised June 30, 2010; accepted July 15, 2010.

This work was supported by the MKE, Rep. of Korea through the ITRC program, [NIPA-2010-(C1090-1011-0005)].

Tran Minh Trung (phone: +82 2 2123 5729, email: trungtm@kaist.ac.kr) is with the Department of Information and Communications Engineering, KAIST, Daejeon, Rep. of Korea.

Jeonghoon Mo (corresponding author, email: j.mo@yonsei.ac.kr) is with the Department of Information and Industrial Engineering, Yonsei University, Seoul, Rep. of Korea.

doi:10.4218/etrij.10.0210.0102

frames of a fixed duration *T*, each of which is subdivided into a set of *k* slots. Each slot is long enough to transmit one or more packets. Hence, node *n* picks one of the *k* slots to transmit a packet.

There is a set *F* of flows, and flow $f \in F$ is specified by a node set $R(f) = \{v_1, v_2, \ldots, v_n\}$, where v_i is the *i*-th node on the route. Here, the first node v_1 is the source node, and v_n is the receiver node. In Fig. 1, there is a flow of which $R(f_1) = \{1, 2, 3, 4\}$. Node 1 is a source node, and node 4 is the destination node. There are three channels, and k=5 slots in a frame.

As a node uses a combination of channel and time slot (*c*, *s*) to transmit a packet, these two parameters need to be determined either online or offline. Hence, there is a channel and time-slot allocation problem for the mesh network. As we are interested in the QoS, our objective is to significantly reduce the delay of flows. In particular, we are interested in finding a suboptimal solution to minimize the maximum delays over flows because this problem is NP-complete. The simple version is shown to be so in [2].

III. Longest Flow First (LFF) Channel and Time-Slot Allocation Algorithms

First, we allocate channels for all links and then determine time slots for a given channel in our algorithm. The key idea is to give higher priority to the longer flows than the shorter ones. This idea is beneficial because, in WMNs, delay and throughput performance degrade significantly with an increase in the number of hops.

1. LFF Channel Allocation

Our algorithm uses the contention degree of secondary conflict links. The contention degree of a link *l* in channel *c* is the number of interfering links in the channel. For example, link (1, 2) has the degrees of 0 and 2 in channels 1 and 2, respectively. The *conflict links* are a pair of links within interference range of each other. They are *primary* if they have a common node; otherwise, they are *secondary*. Links (1, 2) and (2, 3) are primary conflict links, while links (1, 2) and (3, 4) are secondary conflict links.

In LFF channel allocation (LFF-CA), we use ideas of the distance-1 algorithm and further extend it by taking into account the length of flows and utilizing the contention degree or the number of secondary conflict links to assign the channel as follows:

Step 0. Initialize secondary contention degrees to be 0.
Step 1. Sort flows in a descending order of lengths.
Step 2. Select the longest length flow *f* among flows that are not allocated yet.

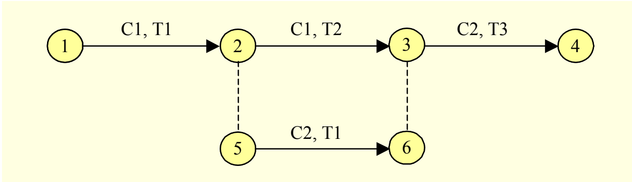

Fig. 1. Example of conflict free channel and time-slot assignment for links in two flows, $R(f_1)$={1, 2, 3, 4} and $R(f_2)$={5, 6}.

Step 3. For all links *l* of flow *f* that are not assigned:
- Assign channel *c* with minimum contention degree.
- If there is a tie, and if the contention degree of the previous link is the minimum, assign the channel of the previous link.
- Otherwise, randomly select one among the minimum contention degree channels.
- Update secondary contention degrees.

Step 4. If *l* is the last link of flow *f*, go to step 2; otherwise, go to step 3.

Consider the example shown in Fig. 1 with two flows, $R(f_1)$ = {1, 2, 3, 4} and $R(f_2)$ = {5, 6}. Assume that there are two available channels. The LFF-CA algorithm starts assigning channels for links of f_1, as it is the longest. For link (1, 2), a channel is randomly selected because the contention degrees are initialized as 0 at the start. Assume that channel 1 is assigned. Then, the algorithm assigns the same channel to link (2, 3) because of the precedent link rule. Now, the contention degrees of link (3, 4) on channels 1 and 2 are 1 and 0, respectively. Note that we are counting the number of secondary conflict links that do not have common nodes with (3, 4). Thus LFF-CA assigns link (3, 4) to channel 2, on which link (3, 4) has the lowest contention degree. Because all links of f_1 are allocated, LFF-CA selects the next flow f_2 with one link (5, 6). Because the contention degrees of this link on channels 1 and 2 are 2 and 1, respectively, channel 2 is assigned.

2. LFF Time-Slot Allocation

Time-slot allocation in WMNs was proposed in [2] to minimize end-to-end delay in a tree-based topology. However, it is limited to a network with a single channel and a single gateway. We propose an algorithm called LFF time-slot allocation (LFF-TA) that works for multiple channels with multiple gateways. Like the channel allocation algorithm, it utilizes flow length information.

The basic idea is to allocate the time slots of a flow so that a packet of the previous link can be transmitted immediately after completion. For example, if time slot 1 is used for flow 1 in the previous link, allocation of time slot 2 or a slot near 2 can streamline the transmission. We give priority to links of longer

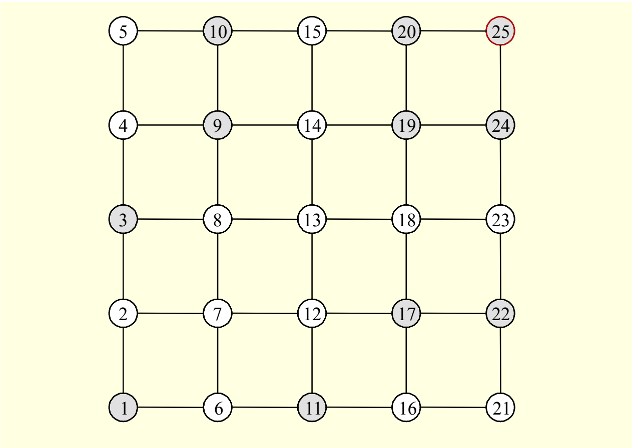

Fig. 2. Grid scenario: 5×5 size.

flows as in LFF-CA by allocating their time slots earlier than those of shorter flows. Each link maintains a list of available time slots that are not used by conflicting links, and one of them is selected accordingly. The process is carried out as follows:

Step 0. Initialize available time slots to be {1, 2, 3,…, T}, where T is the number of time slots.

Step 1. Sort flows in descending order of lengths.

Step 2. Select the longest flow f among flows that are not allocated yet.

Step 3. For all links l of flow f that are not assigned:

- Assign time slot t that is behind and closest to the time slot of the precedent link among available time slots.
- If no such slot exists, increase T by one and restart the algorithm.
- Update available time slot lists.

Step 4. If l is the last link of flow f, go to step 2; otherwise, go to step 3.

Figure 1 shows the results of LFF-TA. In the longest flow f_1, link (2, 3) is allocated to transmit in time slot 2, immediately after the previous link (1, 2), which is allocated to transmit in time slot 1. Since two secondary conflict links, (5, 6) and (1, 2), are allocated on different channel, they can be scheduled to transmit in the same time slot 1.

The proposed algorithms are semi-static in that we assumed a given set of flows over a given topology. The channel and time-slot allocations can be changed at every fixed interval as needed. Moreover, they are different from previous algorithms in that they use layer 3 flow information. Most existing channel allocation algorithms are layer 2 algorithms.

IV. Performance Evaluation

The performance of the proposed algorithms was evaluated using a MATLAB simulator that we developed. We simulated the 802.11 scenario with the grid topology shown in Fig. 2. The simulation parameters are shown in Table 1.

Table 1. Simulation parameters.

Parameters	Values
Data rate	2 Mbps
Codec	G729A
Packet size	864 bits
Time-slot size	432 µs
Packet interval	24 slots
Routing protocol	Load balancing, Min-hop

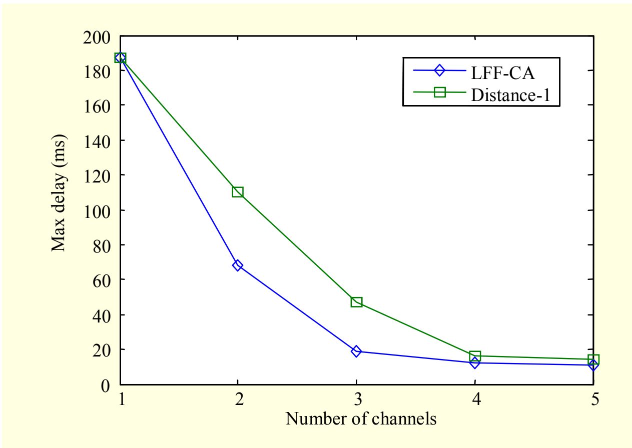

Fig. 3. Delay (LFF-CA vs. distance-1).

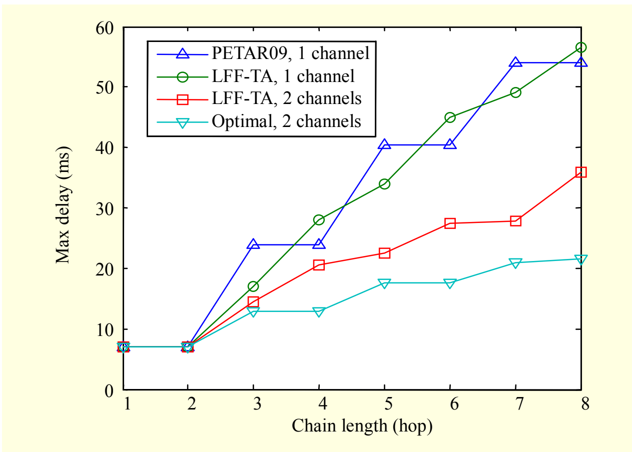

Fig. 4. Max-delay of LFF-TA vs. PETAR09 [3] and optimal.

1. LFF-CA versus Distance-1 CA.

We generated 10 bidirectional voice calls from a set nodes (1, 3, 10, 20, 11, 22, 24, 9, 19, 17) to the gateway node 25. The best-effort data traffic was generated from all nodes at a rate 2 Mbps, which could be transmitted only when the time slots were not being used for voice traffic. Figure 3 shows the delay performance of the two-channel allocation algorithm for various numbers of channels from 1 through 5. The max-delay was reduced by as much as 60% with 3 channels. The difference is small when the number of channels is high, because 4 channels are sufficient for conflict-free scheduling in grid topology [3].

2. LFF-TA

We compared the LFF-TA with other algorithms, such as PETAR09 [2]. The simulation was done in a linear topology with 1 and 2 channels. The length of the chain was varied from 1 to 8. Figure 4 shows the delay performance of PETAR09 and LFF-TA with one and two channels. The optimal solution was obtained from full enumerations. As PETAR09 was designed for only single-channel networks, it achieves similar performance to that of LFF-TA using a single channel. When LFF-TA uses two channels, the max-delay is reduced by as much as 48% in comparison with single-channel performance.

V. Conclusion

In this letter, we introduced McTMAC protocol that can efficiently handle the delay over multihop WMN. We extended the time-slot assignment of the PETAR09 algorithm to multichannel time-slot allocation and enhanced the distance-1 channel assignment. The simulation results demonstrated that our time-slot algorithm achieves a result similar to that of PETAR09 in single-channel time-slot allocation and outperform it by using multichannel. It also shows that, in some cases, the max-delay is significantly reduced when LFF-CA is used instead of distance-1 CA. More simulation results and details such as implementation issues can be found in [6].

References

[1] IEEE, "802.11 TGs MAC Enhancement Proposals," Protocol Proposal IEEE 802.11-05/0575r3, 2005.

[2] P. Djukic and S. Valaee, "Delay Aware Link Scheduling for Multi-Hop TDMA Wireless Networks," *IEEE/ACM Trans. Netw.*, vol. 17, no. 3, June 2009, pp. 870-883.

[3] E. Aryafar, O. Gurewitz, and E. Knightly, "Distance-1 Constrained Channel Assignment in Single Radio Wireless Mesh Networks," *IEEE INFOCOM*, Apr. 13-18, 2008, pp. 762-770.

[4] H. Yu, P. Mohapatra, and X. Liu, "Channel Assignment and Link Scheduling in Multi-Radio Multi-Channel Wireless Mesh Networks," *Mob. Netw. Appl.*, vol. 13, Apr. 2008, pp. 169-185.

[5] M. Yun et al., "Channel-Assignment and Scheduling in Wireless Mesh Networks Considering Switching Overhead," *IEEE ICC*, June 14-18, 2009, pp. 1-6.

[6] T.M. Trung, "A Multi-Channel TDMA MAC Protocol for Supporting QoS over Multi-Hop Wireless Mesh Networks," PhD Thesis, KAIST, E-library, 2009.