An Adaptive Multi-channel MAC protocol for Wireless Ad Hoc Networks

Wen-Tsuen Chen* Ting-Kai Huang* Yu-Chu Chang** Jen-Chu Liu*

*Department of Computer Science
National Tsing Hua University
Hsin-Chu, Taiwan 30043
{wtchen, mr924325, dr888301}@cs.nthu.edu.tw

**Institute of Communication Engineering
National Tsing Hua University
Hsin-Chu, Taiwan 30043
dr939607@com.nthu.edu.tw

Abstract — **Employing multiple channels is an effective way to improve the performance of wireless networks [2-10]. Some previous works on multiple channels [2-4] assume at least two transceivers and result in a higher cost on hardware requirements. The protocol in [5] uses a single transceiver and divides the beacon interval into two parts: channel negotiation and data transmission. However, the fixed length of channel negotiation interval limits the channel utilization. This paper proposed a new single transceiver MAC protocol that can exploit multiple channels effectively and dynamically adjust the length of negotiation interval to better utilize the wireless channel resources. The simulation results show that the proposed protocol achieves higher aggregate network throughput than the fixed schemes on various traffic loads.**

I. INTRODUCTION

A mobile ad hoc network (MANET) consists of mobile hosts, which can transmit messages directly to each other within their radio transmission range without any infrastructure devices. In addition, they can send the messages to distant destinations with other intermediate hosts to relay the messages. Recently, there are extensive studies on routing and medium access control protocol for MANETs. The bandwidth of wireless links is so precious that there are number of medium access protocols [14, 15] proposed to maximize the utilization of bandwidth. However, it is limited for improving the throughput of the entire network because of the potential bandwidth of one channel. In order to increase the entire throughput of a network, it is a fundamental and effective way by using multiple channels simultaneously.

IEEE 802.11 standard [1] provides multiple channels for wireless communication. For example, 802.11b has 14 channels and 802.11a has 12 channels (8 channels for outdoor use, and 4 channels for indoor use). However, in practice, only one channel is in-used at any one time. When a wireless device is switched on, it first searches for the channel with best quality available and uses that channel. By exploiting multiple channels available at the same time, we can achieve higher throughput than just using one channel. The prerequisite is that the frequency spacing must be at least 30 MHz so that each channel does not interfere with each other.

It is a challenging problem to design a MAC protocol for exploiting multiple channels because of characteristic of wireless devices nowadays. They are often equipped with a single half-duplex transceiver. Although they can switch between the available channels, they can only listen to one channel at the same time. This is the main reason that a multi-channel MAC protocol is so difficult to design. The traditional 802.11 MAC protocol designed for a number of mobile hosts that share one channel is not suitable for a network with mobile hosts sharing multiple channels.

In this paper, an adaptive multi-channel MAC protocol is proposed for MANETs. We divide available channels into two groups, one for transmitting control messages and the others for data messages. Resembling ideas of separating available channels into control channels and data channels have been proposed before. However, in this research, we further segment a time frame into two intervals: negotiation interval and data transmission interval. In negotiation interval, two communicating hosts negotiate with each other to reserve an appropriate channel for data transmission. By this way, each host can exploit multiple channels effectively by using only single transceiver. In the proposed protocol, the length of the negotiation interval and the data transmission interval in a time frame can be dynamically adjusted according to network traffic condition to maximize the channel utilization.

The rest of the paper is organized as follows. Section 2, we simply review the related researches in this area. In Section 3, the proposed Multi-channel MAC protocol is presented in detail. Simulation results are shown in Section 4. Finally, Conclusions are drawn in Section 5.

II. RELATED WORKS

In a multi-channel wireless ad hoc network, each pairs of sender and receivers may use different channels for exchanging messages. The Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) protocol normally designed for single channel environment does not work effectively in this environment. If a mobile host cannot sense all the channels simultaneously, it may loss the channel reservation information that its neighbors announce on another channel, which is different from the channel it is listening on. Therefore, collisions can still occur easily. This is the multi-channel hidden terminal problem as addressed in [5]. Nasipuri et al. [2] proposed that each mobile host is equipped with multiple transceivers as many as the number of the channels. Then, hosts can not only sense all the channels simultaneously but also use CSMA/CA protocol to avoid collisions. These methods are not practical because of the

This research was supported by the National Science Council, Taiwan, R.O.C. under Grant NSC 93-2213-E-007-002

1-4244-0355-3/06/$20.00 (c) 2006 IEEE

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

expensive hardware costs incurred by multiple transceivers. In order to reduce the hardware costs, [3, 4] proposed that a mobile host need only be equipped with two transceivers, one for control channel and the other for data channels, and can still avoid collisions effectively.

With these approaches, a mobile host exchanges control messages with its destination on the control channel to choose a common channel and reserve it. Then, they will transmit data packets on that channel. Wu et al. proposed a multi-channel protocol, called dynamic channel assignment (DCA) [3] based on these ideas. They assume that every host is equipped with two transceivers. One is dedicated for monitoring the control channel, and the other is responsible for data sending/receiving. Every host maintains its own free channel list (FCL). When a sender A has data to send to its destination B, it reserves a data channel by sending a Request to Send (RTS) carrying its FCL to receiver B. B will choose a free channel that is listed in both the FCLs of A and B as the data channel. The selected channel is piggybacked on the Clear to Send (CTS) message sending back to A. Then sender A will send a Reserved to Send (RES) message to inhibit its neighbors from using this channel. Wu et al. proposed a dynamic channel assignment protocol with power control (DCA-PC) [4] that is a variant of DCA. In DCA-PC, a free channel with least interference is selected. The transmitting power of a host can be reduced to mitigate interference with its neighbors.

Some MAC protocols [5-10] for multi-channel MANET take the hardware costs, energy consumption, and feasibility into consideration. In these protocols, each host uses only one transceiver to achieve multi-channel transmission. There is one channel, which is selected to be the control channel, and the others are as data channels. Furthermore, they divide a beacon interval into two pieces to avoid the multi-channel hidden terminal problem. One is for channel negotiation and the other is for data transmission.

In multi-channel MAC (MMAC) protocol [5], there is a small window at the start of each beacon interval for negotiating the data channel to be used for packet transmissions during this interval. A similar approach is used in the power saving mechanism (PSM) of the IEEE 802.11 standard. The fixed window is called ad hoc traffic indication message (ATIM) window. A fixed window size of the ATIM may limit the performance of networks in MMAC and the PSM in IEEE 802.11 standard. Because of the fixed window size, hosts have to wait until the end of ATIM window in order to transmit data packets even though they have announced all ATIM messages to different destinations when the traffic load is low. The channel utilization is not efficient in this situation. Therefore, we make the window size dynamically adjusted to the traffic load in order to enhance the overall network performance.

III. PROPOSED MULTI-CHANNEL MAC PROTOCOL

This section presents the proposed Traffic Aware Multi-channel Medium Access Control (TA-MMAC) protocol. One of the multiple channels is used as the control channel and the rest are used as data channels. Hosts exchange control messages on the control channel and transmit data packets on a dynamically selected data channel. Time is divided into a number of time frames of fixed size, each time frame preceded by a beacon. Each time frame in TA-MMAC is divided into negotiation interval and data transmission interval. The innovation of our protocol is that the length of the negotiation interval and the data transmission interval in a time frame can be dynamically adjusted according to network traffic condition to maximize the channel utilization.

In each time frame, two communicating hosts negotiate on the control channel to select an appropriate data channel for data transmission and the data channel will be released at the end of this time frame. Each host maintains the status of channel usage for channel selection. For the purpose of channel negotiation, we use the reserved bits of RTS/CTS to contain the channel usage status and some controlling information. These modified RTS/CTS messages are called MRTS and MCTS. The procedure of channel negotiation and data exchange in TA-MMAC is illustrated in Fig. 1. When sender A is to communicate with receiver B, they will go through a MRTS/MCTS/RRTS dialogue, as the three-way handshake, to select a channel for data transmission. First, host A sends a MRTS message, which carries the channel usage status of host A, to host B through the control channel. According to the channel selection rule described in subsection III-A, host B chooses an appropriate channel and notifies host A in the MCTS message. Then sender A broadcasts a RRTS message to announce the reservation to its neighbors. Finally, both sender A and receiver B switch to the selected data channel in the data transmission interval and start data transmission.

A. Channel selection

As shown in Fig. 1, during negotiation interval to reserve a data channel, the status of channel usage of the sender and the receiver should be maintained. Each host maintains an *In-use channel* and two channel lists, *Free channel list* and *Busy channel list*, to keep track of necessary information for channel selection. The *In-use channel* of the host is the channel selected for data transmission in the current time frame. If any other host communicates with this host, it has to use the same channel. *Free channel list* of the host is the list of channels that are not used by any of its neighboring hosts. *Busy channel list* of the host is the list of channels selected as the In-use channel at least by one of its neighboring hosts. There is also a counter for each channel in Busy channel list to count the number of sender-receiver pairs, which are using this channel for data transmission.

According to the channel usage information, host B attempts to select a data channel with less interference. It compares the In-use channel of host A with its In-use channel. When the In-use channels of host A and B are different, it means that both host A and B have reserved their In-use channels for data transmissions with others. Host B will ignore the MRTS

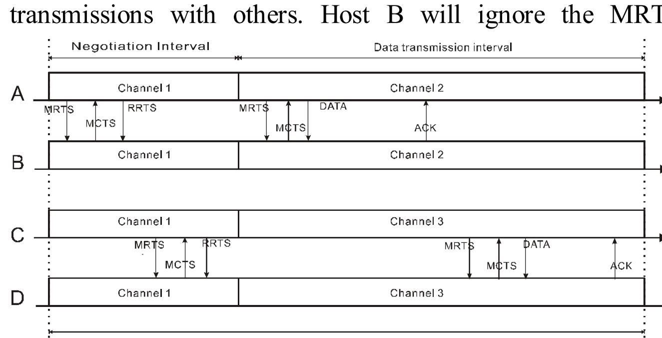


Figure 1. Process of channel negotiation and data exchange in TA-MMAC

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

message and defer the communication with host A. If host A and host B have the same In-use channel, this channel is selected as the data channel to be used in the current time frame. When there is only one host, either host A or B, has an In-use channel, the In-use channel is selected as the data channel for both hosts.

If neither host A nor B has any In-use channel, host B tries to find a common channel that is listed in both the Free-channel lists of host A and host B. If there exists at least one common channel in the Free channel lists, host B randomly chooses one of them as the data channel. Otherwise, the channel that is in the Free channel list of one side and on the Busy channel list of the other side with least value of its counter will be selected as the data channel.

Finally, if host B cannot select an appropriate channel from In-use channels or from the Free-channel list, it means that there is no channel, with which is not interfered by all neighboring hosts. In order to accommodate the hosts around both host A and host B, a channel on both Busy-channel lists of host A and B is selected according to the sum of the value of the counters. The channel with the least sum would be selected since that it would encounter less contention among hosts.

As the analysis in [13], the probability of a packet collision in IEEE 802.11 MAC protocol depends on the number of hosts in a network and the minimum contention window size. This channel selection algorithm attempts to balance the channel load as much as possible, so that the bandwidth wastage caused by contention and backoff is reduced. For this reason, we count the number of sender-receiver pairs that have reserved the channel and use this information as one of the criteria for selecting an appropriate data channel. This scheme assumes that each pair of sender and receiver has the same amount of traffic, which is not true. Counting the number of packets that would be transmitted in the current time frame on the channel would be better.

B. Dynamic Interval Adjustment

In order to fully utilize the channels in each time frame, the lengths of negotiation and data transmission intervals should be dynamically adjusted according to the network traffic load. With a small negotiation interval in case of heavy traffic loads, most hosts may not successfully exchange MRTS/MCTS/RRTS messages with their receivers, so they may have no chance to transmit data packets in the following data transmission interval. On the other hand, if the traffic load is light and the negotiation interval is set too long, the channel utilization is inefficient because all sender-receiver negotiations would be finished before the end of the interval. Therefore, dynamically adjusting the ratio of the two intervals according to network conditions is essential for improving network performance.

The mechanism of dynamic interval adjustment is to optimize the negotiation interval size so that every host has an opportunity to make negotiation once in a time frame. We use an opportunistic method that if any host needs to negotiate with someone, it sends a request to borrow some negotiation time from its neighbors. When the host finishes its work, it gives the time back to its neighbors.

In order to prevent the extreme condition that may cause the adjustment mechanism working poorly, there are preset minimum and maximum values for the negotiation interval size. The minimum negotiation interval size is to prevent the negotiation interval from being adjusted too small a value that there is not any host can announce the negotiation request. On the other hand, the preset maximum value limits the negotiation interval so that a host may have data transmission time. The increment or decrement of the negotiation interval is a multiple of level of fixed size. Let L be the size of one level. The transmission times for MRTS, MCTS, and RRTS are T_{MRTS}, T_{MCTS}, and T_{RRTS}, respectively. So

$$L = DIFS + averageCW + T_{MRTS} + SIFS + T_{MCTS} + SIFS + T_{RRTS} \quad (1)$$

where $averageCW = slotTime*(CW_{min}-1)/2$ is the average backoff time when there are no other contending stations. We ignore the possibility of collisions and the increase of backoff time in subsequent retransmission after a collision here.

The negotiation interval sizes of all hosts are initially set to a default minimum. A Host will announce an increase or decrease request to ask its neighbors to adjust their negotiation interval sizes by one level (the size of one level is defined in (1)) each time when meeting one of the adjusting rules. It is important that only the hosts, which did not try to send packets in the previous first time frame, can announce an increase request according to the increase rules. On the other hand, the hosts can announce a decrease request if they have transmitted data packets in the previous time frame. The adjusting rules are as follows:

1) Increase Rules: A host asks its neighbors to increase the size of negotiation interval when it cannot announce a negotiation request in the last time frame. The request is included in each negotiation messages. Hosts, receiving or overhearing this kind of negotiation messages, increase their own negotiation intervals at the next time frame. Each host, which does not transmit data packets in the previous two time frames, can announce the increase request once at most in a time frame. As an example shown in Fig. 2, host C sends a negotiation request to host D unsuccessfully because of sensing the communication of host A and host B. At the next time frame, host C gets a chance to make a negotiation with host D successfully and asks other hosts to increase their negotiation interval. Then, each host increases its own negotiation interval at the third time frame. Two pairs of communication hosts, host A, host B, host C, and host D, can make their data transmission successfully and simultaneously.

A host also includes an increasing request in negotiation messages when it senses that the idle time of the negotiation interval in current time frame is not longer than a particular length, as long as one adjusting level. In this case, it seems to be that the negotiation interval is large enough for all necessary negotiations because the host can successfully negotiate with its destination. However, the crowded negotiations messages show that the negotiation interval is not large enough for all hosts, which are trying to transmit data. For this reason, the host should broadcast an increase request.

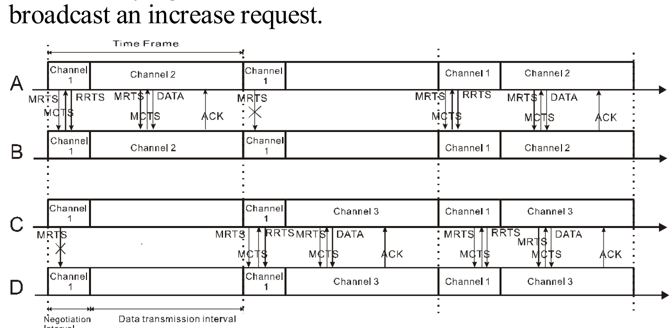


Figure 2. Example of Increase Rule.

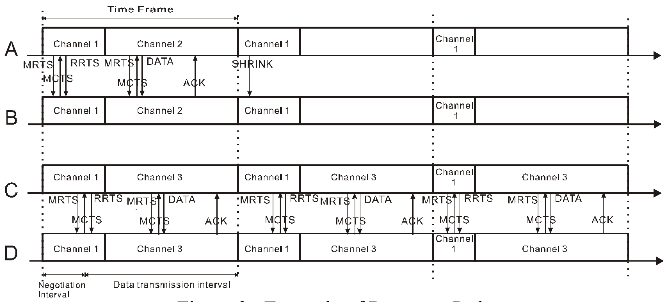


Figure 3. Example of Decrease Rule

2) Decrease Rule: A host releases the negotiation time that is borrowed from other hosts when it finishes its work. The host broadcasts a new type message, SHRINK, to tell other hosts to decrease their negotiation intervals by one level when it finishes its data transmission and does not have any packet in the next time frame. By this way, hosts try to optimize the size of their negotiation intervals to fit the requirements. If the negotiation interval is set too large, hosts are idle before the end of the interval. Hosts get longer data transmission time by shrinking the negotiation interval. In Fig. 3, host A finishes its transmission with B at first time frame, so it broadcasts a SHRINK message to ask other hosts to decrease their negotiation intervals by one level. At the third time frame, each host decreases its negotiation interval and can make more data transmission.

In order to further improve the performance of the proposed TA-MMAC, a sender can issue multiple negotiations, each to a different receiver, in a single time frame as long as the aggregate length of data packets do not exceed the size of data transmission interval. When host A wants to communicate with host B, host A predicts the total transmission time needed for transmitting data. If the amount of time is less than one time frame, host A can make reservation with other destinations. Otherwise, if host A cannot finish the data transmission with host B in the current time frame, host A will tell host B to extend the data transmission to next time frame by notifying host B the number of packets pending for it. In TA-MMAC, a host can extend its data transmission time to the next time frame at most. After that, if the host cannot finish the transmission, it should re-negotiate with its receiver, as usual.

Moreover, each host negotiates with one destination once in a time frame in TA-MMAC and a successful negotiation allows data transmission to cross over at most two time frames for favoring large packet transmissions between a pair of sender and receiver. In this way, each host generates fewer negotiation requests in a negotiation interval; decreases its negotiation interval without degrading success rate of negotiations. Therefore, more bandwidth can be dedicated to data transmission.

IV. Simulation and Discussion

In this section, we firstly try to find out the optimal size of negotiation interval at different network traffic load. Then, we simulate the proposed TA-MMAC protocol, the MMAC protocol, and Dynamic MMAC (D-MMAC) protocol. Jung et al. proposes a method to dynamically control the ATIM window in 802.11 power saving mode [12]. They also use a heuristic method to adjust the window size according to different network traffic load. Hosts adjust their own ATIM windows independently when meeting the defined four increase rules and one decrease rule. According to their simulation results, their dynamic mechanism can make comparable or better performance than IEEE 802.11 without any nodes in power saving mode in throughput. For this reason, we implement this method on MMAC protocol to adjust the negotiation interval size with little adjustment for multi-channel environment, and call it as D-MMAC. The simulation results, in Fig. 5 and 6, show that the mechanism is work and gets higher throughput and lower latency than original fixed MAC protocol.

We use two metrics to evaluate performance of the proposed protocol.

1) Aggregate throughput over all flows in the network: Our protocol is expected to increase the total throughput of network. It is supposed to perform better than MMAC by dynamically adjusting the size of NTI to the traffic load of network. This metric directly shows how our protocol achieves this goal.

2) Average packet delivery delay over all flows in the network: The packet delay is the duration between the time when the link layer of the sender receives a packet to send, and the time the packet reaches the destination. So the average packet delivery delay is the sum of delays for queuing, backoff, channel negotiation and transmission delay. One of our goals is to exploit the multiple channels more efficiently than MMAC without increasing the packet delivery delay. Therefore, the metric is useful to measure the delay increasing in the proposed protocol.

A. Simulation Model

We use C-SIM [11] to implement a simulator to compare the performance of the proposed TA-MMAC protocol, the MMAC protocol, and the D-MMAC protocol. The duration of each simulation is 10 seconds in a mobile ad hoc network. In each case, half of the hosts are sources and the rest are destinations, for the simulated flows. For example, in the 8 hosts scenario, 4 hosts send packets to the other 4 hosts. Each flow transmits Constant Bit Rate (CBR) traffic. The parameters we vary include the number of hosts in the network, the network traffic load, and the negotiation interval size of each host. All hosts are within each other's transmission range. The other parameters are listed in table I.

B. Optimal Negotiation Interval Size Evaluation

We change the number of hosts in the network and the network traffic load to find out the optimal size of negotiation interval. The simulated traffic load is a fraction of aggregate bit

TABLE I
Simulation Parameters

Parameters	Values
Length of time frame	100 ms
Number of channels	3
Bandwidth of channel	11 Mbps
Packet size	512 bytes
Max negotiation interval size	26 ms
Min negotiation interval size	5 ms
Slot time	0.02 ms
Length of MRTS	20 bytes
Length of MCTS	14 bytes
Length of RRTS	14 bytes
Length of SHRINK	14 bytes

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

rate of available channels. For example, there are 3 channels in the network, and the bandwidth of each channel is 11 Mbps. The network traffic load of 10% is 11*3*0.1=3.3Mbps.

Fig. 4 shows the aggregate throughput, with different fixed negotiation interval sizes of different network traffic load when the numbers of hosts are different. In Fig. 4, we find that the optimal size of negotiation interval correlates with the number of hosts and the network traffic load. We fix the network traffic load, and change the number of hosts to evaluate the optimal size of negotiation interval. The optimal value is changed according to different number of hosts. On the other side, if we fix the number of hosts in the network and use different traffic load for test, there are also different optimal negotiation interval sizes in different cases.

The evaluation results show that if we can get the information of the number of hosts in the network and the network traffic load fast and entirely, we can predict the optimal value of negotiation interval size. However, it is hard to achieve in a mobile ad hoc network. Each host dynamically joins the network, and has different traffic requirement at different time. Since it is hard to collect the necessary information, such as the number of hosts in the network and the traffic requirement of each host, we propose a heuristic method to adjust the negotiation interval size. The simulation results shown in the next subsection also demonstrate that our method can adjust the negotiation interval size appropriately to improve the aggregate throughput in the network.

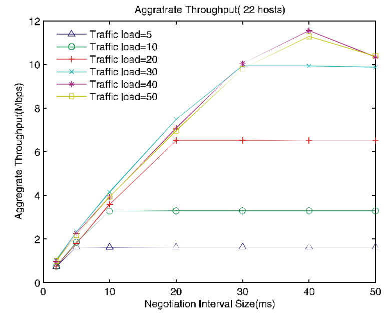
C. Performance Evaluation in TA-MMAC, D-MMAC, and MMAC

Fig. 5 shows the aggregate throughput (aggregate throughput over all flows) with CBR traffic when using MMAC with different size of negotiation interval, D-MAC, and TA-MMAC schemes. The results in Fig. 5(a), (b), and(c) are for different CBR traffic of hosts in the network.

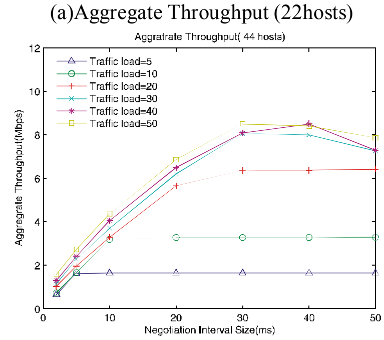
As the figure shows, the size of negotiation interval is correlated with the throughput of MMAC. When the CBR is low, the throughput with MMAC is less sensitive to the size of interval. In Fig. 5(a), the negotiation interval size can be set from 20ms to 50ms to get almost the same aggregate throughput. However, as the CBR increases, the negotiation interval size affects the throughput with MMAC significantly.

On the other hand, TA-MMAC typically performs comparable or better than MMAC or D-MAC. In TA-MMAC, a host makes negotiations with each destination of their pending packets only one time in each time frame. When there are pending packets that are not transmitted, the sender and receiver can continue transmitting packets extending to the next time frame without any additional negotiation. Therefore, TA-MMAC needs fewer necessary negotiations and can have longer size of data transmission interval to transmit data packets. It makes the TA-MMAC perform better than MMAC.

In addition, the TA-MMAC can get higher throughput that D-MMAC because of adjusting interval size more correctly. In TA-MMAC, Hosts increase the negotiation interval size as much as their necessary. Host will ask some negotiation time from other host by sending increase request. When hosts do not have pending data packets, they tell other hosts to decrease their negotiation interval size by broadcasting SHRINK messages. By this way, the negotiation interval is set closed to the optimal size according to the current network traffic situation.



(a)Aggregate Throughput (22hosts)



(b) Aggregate Throughput (44 hosts)

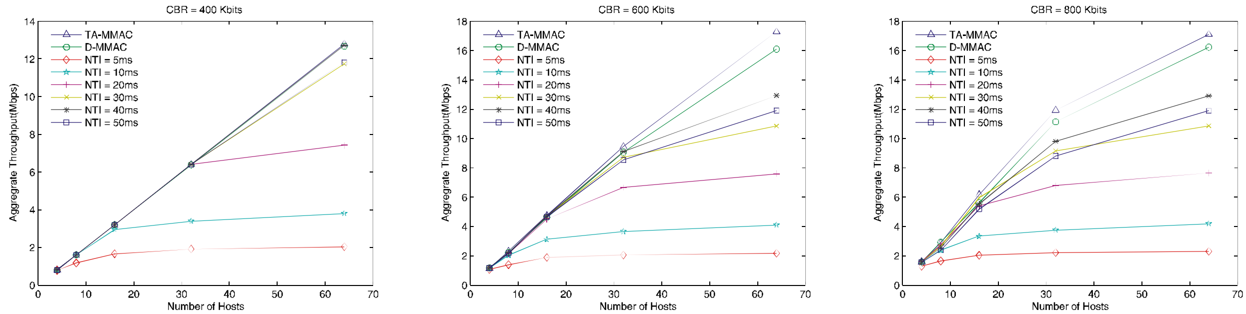
Figure 4. Aggregate Throughput

In Fig. 5, we fix the traffic load of each flow. We can find that the aggregate throughput is low when the size of negotiation interval is too small or large. If the size of negotiation interval is too small, hosts cannot negotiate with their destinations successfully and the throughput degrades. However, when the size of negotiation interval is too large, the hosts do not have enough time to transmit data packets, and the throughput degrades as well.

Fig. 6 shows the average packet delay of the TA-MMAC, D-MMAC and MMAC. When the traffic load is low, there is a wide range of negotiation interval size that can minimize the delivery delay of packets. However, when the traffic load is high, the appropriate range of negotiation interval size is much smaller. If the size of negotiation interval is too large, there will be a lot of idle time that hosts are waiting to transmit data packets. It increases the average packet delivery delay. Nevertheless, fail negotiations make the hosts cannot transmit data packets in this time frame right away affects the latency more significantly than the idle waiting. That is the reason why the latency is much larger with smaller size of negotiation interval than with larger one.

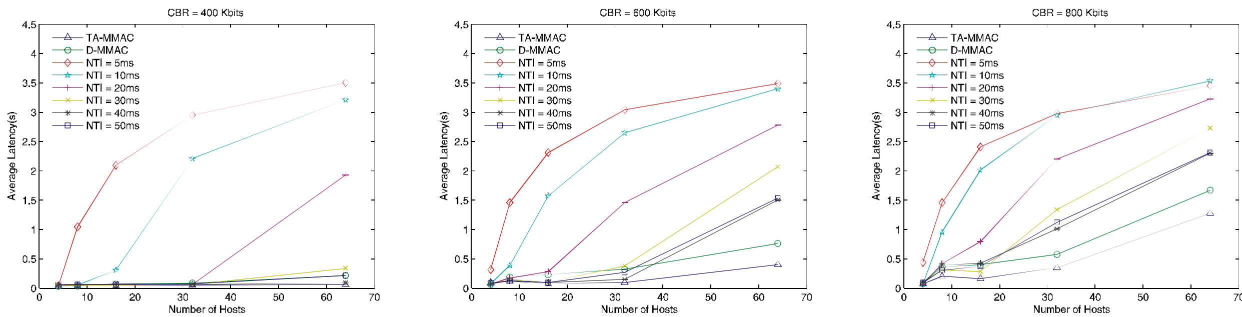
The average packet delay of TA-MMAC is less than MMAC that is using the best size of negotiation interval especially when the number of hosts is large. TA-MMAC reduces unnecessary idle time in the negotiation interval by appropriately adjusting the size of negotiation interval according to the network traffic load. In TA-MMAC, a sender can continue sending data packets to its destination in the next time frame without any additional negotiation if it cannot transmit all the pending packets. It can reduce the size of negotiation interval because of fewer necessary negotiations. Hosts can transmit more data packets because of the longer data transmission time. The right size of negotiation interval makes the higher channel utilization. It is the main reason that the TA-MMAC can have less latency than MMAC.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.



(a)CBR is 400 Kbits/sec (b)CBR is 600 Kbits/sec (c)CBR is 800 Kbits/sec

Figure 5. Aggregate Throughput in MMAC, D-MMAC, and TA-MMAC



(a)CBR is 400 Kbits/sec (b)CBR is 600 Kbits/sec (c)CBR is 800 Kbits/sec

Figure 6. Average Latency in MMAC, D-MMAC, and TA-MMAC

V. CONCLUSIONS

In this paper, we proposed a traffic aware multi-channel MAC protocol, TA-MMAC, which can exploit multiple channels effectively by only using one transceiver per host. In TA-MMAC, we divide time into fixed time frame by using beacon. Furthermore, each time frame is divided into two intervals, negotiation interval and data transmission interval. In negotiation interval, two communicating hosts negotiate with each other to reserve a channel for use during the current time frame.

As the traffic load gets heavier, the desirable size of negotiation interval becomes larger, and vice versa. The use of negotiation interval properly can increase the throughput and decrease the latency. In TA-MMAC, a host can adjust the size of the negotiation interval according to the observed network traffic condition. By dynamically adjusting the size of the negotiation interval, it can allow more negotiations that are successful and get more data transmission time than MMAC. Simulation results show that the proposed scheme can improve the aggregate network throughput and decrease the average latency.

REFERENCES

[1] IEEE 802.11 Working Group, "Wireless LAN medium access control (MAC) and Physical Layer (PHY) specifications," 1997.

[2] Nasipuri, A.; Zhuang, and J.; Das, S.R."A multichannel CSMA MAC protocol for multihop wireless networks," *IEEE Conf. WCNC*, vol.3, pp. 1402-1406, Sept. 1999.

[3] S.-L. Wu, C.-Y. Lin, Y.-U. Tseng, and J.-P. Sheu, "A new nulti-channel MAC protocol with on-demand channel assignment for multi-hop mobile ad hoc networks," *Proc. Int. Symp. Parallel Architectures, Algorithms and Networks,* pp. 232-237, Dec. 2000.

[4] S.-L. Wu, Y.-U. Tseng, C.-Y. Lin, and J.-P. Sheu, "A multi-channel MAC protocol with power control for multi-hop mobile ad hoc networks," *Int. Conf. Distributed Computing Systems Workshop,* pp.419-424, April 2001

[5] J. So, and N. Vaidya, "Multi-channel MAC for ad hoc networks: Handling multi-channel hidden terminal using a single transceiver," *ACM Int. Symp. Mobile Ad Hoc Networking and Computing,* pp. 222-233, 2004.

[6] J. Chen, S.-T. Sheu, and C.-A. Yang, "A new multichannel access protocol for IEEE 802.11 ad hoc wireless LANs," *Proc. IEEE PIMRC*, Vol. 3, pp. 2291-2296, Sept. 2003.

[7] X. Tian, Y. Fang, and T. Ieguchi, "Multichannel time-spread scheduling: A new approach to handling heavy traffic loads in ad hoc networks", *IEEE Conf. WCNC*,Vol. 2, pp. 1075-1080, March 2004.

[8] A. Tzamaloukas, J.J. Garcia-Luna-Aceves,"A receiver-initiated collision-avoidance protocol for multi-channel networks," *Proc. IEEE INFOCOM.*, Vol.1, pp. 189-198, April 2001.

[9] N. Jain, S.R. Das,andA. Nasipuri, "A multichannel CSMA MAC protocol with receiver-based channel selection for multihop wireless networks," *Proc. Computer Communications and Networks,* pp.432-439, Oct. 2001.

[10] P. Bahl, R. Chandra, and J. Dunagan, "SSCH: Slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad hoc wireless networks," *ACM Int. Conf. Mobile Computing and Networking*, pp 216-230, 2004.

[11] Mesquit Software, Development Toolkit for Simulation and Modeling, CSIM, http://mesquite.com/documentation/index.htm

[12] E.-S. Jung, and N.-H. Vaida, "An energy efficient MAC protocol for wireless LANs, "*Proc. IEEE INFOCOM*, Vol.3, pp. 1756-1764, June 2002 .

[13] Y.C. Tay, and K.C. Chua, "A capacity analysis for the IEEE 802.11 MAC protocol," *Wireless Networks*, Vol. 7, pp. 159-171, March 2001.

[14] F. Cali, M. Conti, and E. Gregori, "Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit," *IEEE Trans. Networking*, Vol. 8, pp. 785-799, Dec. 2000.

[15] D. Zheng, and J. Zhang, "Protocol design and performance analysis of opportunistic multi-channel. medium access control," *Proc. IASTED CIIT'03*, 2003.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.