A Reliable Multi-Channel MAC Protocol with Efficient Multicast Support for Ad hoc Networks

Sheng-Hsuan Hsu¹, Ching-Chi Hsu^{1,2} and Ferng-Ching Lin¹

¹Department of Computer Science and Information Engineering, National Taiwan University, Taipei 106, Taiwan
{hsush, cchsu}@csie.ntu.edu.tw

²Kai Nan University, Taoyuan, Taiwan

Abstract - Reliable multicast support at the MAC (medium access control) layer will be of great benefit to the routing function, multicast applications and cluster maintenance in ad-hoc networks. In this paper, we propose a reliable MAC protocol with multicast support for multi-channel ad-hoc networks. This protocol, named SNMS (Sequenced Neighbor with Multicast Support), uses sender-based data transmission strategy in order to provide efficient multicast. SNMS uses one control channel to deliver the control packets and several data channels to delivery data packets. In SNMS, (i) no any handshake (such as CTS/RTS) or any carrier sensing technology is needed, (ii) no hidden terminal problem or exposed terminal problem will exist, and (iii) no form of clock synchronization is required. Compared with SNDR (Sequenced-Neighbor Double-Reservation), the proposed protocol achieves higher reliable multicast efficiency; namely, it delivers multicast packets to receivers reliably with a smaller number of transmissions. Simulation results show that, in the multicast scenario, SNMS delivers a significant improvement over to SNDR.

Keywords: Ad hoc network; broadcast; medium access control (MAC); multi-channel; reliable multicast.

1. INTRODUCTION

Many MAC (*medium access control*) protocols have been developed for wireless networks [1] [3] [4] [5], which assume a common channel shared by mobile hosts in the network. We call these protocols *single-channel* MAC protocols. IEEE 802.11 [7], a standard that has been widely accepted, is also a protocol based on the single-channel model. Due to higher contentions and collisions, the performance of single-channel MAC protocols decrease quickly while the number of mobile hosts increases. To relieve this problem, utilization of multiple channels is one approach. Cai and Lu proposed a MAC protocol, termed SNDR [2] (Sequenced Neighbor Double Reservation), for multi-channel ad-hoc networks. SNDR uses the neighbor-sequenced method to avoid contentions and the double reservation method to improve the total throughput of ad-hoc networks.

Multicast has attracted considerable interest in ad-hoc networks. Usually, the multicast problem is resolved at the network layer. These approaches are utilized above MAC

protocols. Nevertheless, many MAC protocols, such as IEEE 802.11, lack explicit support of reliable multicast or broadcast. To reliably transmit a packet to several neighbors in such MAC protocols, a node must transmit the same multicast packet multiple times, once to each multicast-group neighbor. However, this approach is not scalable since the time to complete a multicast increases with the number of the receivers. Tang and Gerla [6] proposed a reliable broadcast MAC protocol based on IEEE 802.11. However, the proposed protocol was available on the signal-channel model.

In this paper, we propose a reliable MAC protocol, named SNMS (Sequenced Neighbor Multicast Support), can support reliable multicast for multi-channel ad-hoc networks. On the multi-channel model, SNMS can avoid hidden terminal problem and exposed terminal problem. In addition, the mobile hosts will not be interfered while their neighbors are sending or receiving the multicast packets. SNMS can make use of broadcast nature of wireless medium efficiently and still broadcast reliably. Simulation results show that, in the multicast scenario, SNMS delivers a significant improvement over SNDR.

2. SNMS PROTOCOL

2.1. Preliminaries

In our model of the wireless ad-hoc network, nodes are distributed randomly over a two-dimension plane. Assume there are N mobile nodes in the ad-hoc network. All mobile nodes have the same transmission range. We call the set is a neighbor set of a node if this set includes all neighbors of the node. In ad-hoc networks, mobile nodes can randomly move at any given time, hence the neighbor sets are dynamic. In SNMS, we assume that a neighbor set can be correctly updated at any given time and it will keep unchanged during the transmission of one MAC frame. Every node has one antenna with half-duplex operation. It means that the mobile nodes can not receive the data while sending the data nor send the data while receiving the data, even if receiving and sending are in the different channels (the code sequences for CDMA or the frequencies for FDMA). In our design, every mobile node is assigned a channel to send data packets. To

simplify the problem, we assume that the channels of any two nodes are different and have been successfully pre-assigned in the network. We adopt the sender-base strategy to transmit the data packets in SNMS. If a sender wants to send the multicast packet to some of its neighbors, the neighbors must receive the data via the channel pre-assigned to the sender.

TABLE I
THE FORMAT OF NEIGHBOR TABLES

Neighbor's id	Channel number of the neighbor	Neighbor's Sequence number	Sequence number to the neighbor
...	...	...	...
...	...	...	...

2.2. The Neighbor-Sequenced Scheme

An important basis of our method is the neighbor-sequenced method. The method can identify the neighbor set of a node, assign every neighbor a unique sequence number (to the sender) and maintain the sequence number structure. The neighbor-sequenced process that we use was proposed by Cai & Lu [2]. We describe the process briefly in the following.

The neighbor-sequenced scheme maintains the sequence number structure which indexes the neighbors of node i from 1 to k_i , where k_i is the number of the neighbors of the node i . One table is maintained by each node to record the respective sequence numbers to all neighbors of the node. For example, if node j 's sequence number to node i is 3, that means node j is the third neighbor of node i . If node i finds a new neighbor in its neighborhood, it will assign a sequence number k_i+1 to the new neighbor. (Where k_i is the number of the neighbors of node i before the appearance of the new neighbor in the neighborhood of node i .)

When node i recognizes that one of its neighbor have moved away, the process will send a DELETE packet to all nodes in its neighbor set. The DELETE packet includes the sequence number of the neighbor moved away. After receiving DELETE packet, the neighbors of node i will check if the leaving node's sequence number is less than the sequence number of their own. If the answer is YES, the neighbors of node i will decrease its sequence number (to node i) by one. Otherwise, it will do nothing. With this mechanism, those nodes belong to the neighbor set of node i will keep their sequence number from 1 to k_i (k_i is the size of the neighbor set of node i .) without two neighbors having the same sequence number. Since every node is assigned sequence numbers by all of its neighbors. Every node will know its sequence numbers to all of its neighbors. A neighbor table was constructed to include a sequence number and a channel number with respect to each neighbor, and a sequence number to the neighbor. Table I shows the format of the neighbor tables. Mobile nodes will maintain the

Control Channel

Fig. 1. The structure of the frame in SNMS

sequence structure while it finds out their neighbor sets are changed or the structure is uncorrected.

2.3. The Frame Structure of SNMS

In SNMS, one control channel is utilized for the declaration of multicast. Every mobile node is pre-assigned a data channel that is used to reserve the multicast, transmit data packets and send acknowledgements. Time Division Duplex (TDD) is assumed for the data channels. A neighbor table and a transmission queue are given to every node. Due to the half-duplex property, a node cannot send data while receiving. There are three states in our design: the first is the *receiving state*; the second is the *sending state*; and the third is the *idle state*. When a node has data to send, it will enter the sending state. A node is in the idle state when it is not sending or receiving data. While nodes are in the idle state, they listen on the control channel until receiving the frame start field from some sender or changing into the sending state. If the node is ready for receiving data, it enters the receiving state. Mobile nodes cannot be always in the sending state. They will also prepare for receiving data from its neighbors. As the rule, repeated sending states should be followed by some idle states because if all nodes are in the sending state, the whole network will be in deadlock. In two cases, a node needs to decide which state to enter. One is when the node is in the idle state and it has packets to send. At this time, the node should determine to enter the sending state or stay in the idle state. The other case is when the node is in the idle state and it receives the frame start field which contains its own id. In the later case, the node will determine to enter the receiving state or stay in the idle state. The

structure of the frame in SNMS is shown in Fig. 1. We describe the structure of the frame below:

1. Part A is the *frame start field*, which includes the identity of the multicast sender and the receiver list. The receiver list consists of the sequence numbers (to the sender) of the receivers. The nodes in the idle state will listen on the control channel. After a node received the frame start field from the control channel and found itself in the receiver list, it can process to receive the multicast packet. The length of Part A is $D+\log N$ bits, where D is the maximum number of neighbors that a node can have in the ad hoc network and N is the number of mobile nodes in the network.
2. Part B is the *reserve field* which reserves one bit for each receiver in the order of the sequence numbers to the sender. If a neighbor is ready to receive the data, it will switch to the sender's channel and then set its reservation bit to 1. If the neighbor is busy receiving or sending the data, it means that the neighbor cannot hear the signal on the control channel. No operation is needed since the original value of the reservation bit was 0. For example, if Part A is {0011, 100010}, where 0011 is the sender's id and 100010 is the receiver list. It means that node 3 wants to send the packet to its first neighbor and fifth neighbor. In this case, the length of Part B is 2 bits. The first bit of Part B is for its first neighbor and the second bit is for the fifth neighbor of the sender.
3. Part C is the *control field* that is used for the sender to send the other control information and data sending signal to its neighbors. If there exists any reservation bit (in Part B) set by 1, a *data-sending signal* will be send by the sender to inform the receivers the beginning of the data transmission. If all the reservation bits received by the sender are 0, that is to say, none of the receiver can participate the transmission. Then the sender will terminate the frame of the transmission by sending the *frame-abort signal*. This approach prevents the waste of time slots when none of the receivers can receive the data.
4. Part D is the *data field*. At this part, the sender starts to transmit data and all the receivers get the data concurrently. Note that if the frame-abort signal was send by the sender in part C, this part and part E will be eliminated.
5. Part E is the *acknowledgement (ack) field*. The length of this field is the same as the length of Part B. Each bit of the acknowledgement field is used by one receiver to acknowledge the data received. Since each receiver has different sequence number to the sender, the collision of

acknowledgements can be avoided. With this scheme, SNMS can provide reliable multicast service in the MAC layer.

After the sender receives the acknowledgements field (Part E) from some neighbors, it will delete these neighbors from the receiver list and then queues the unfinished multicast packet in the transmission queue. After all neighbors in the receiver list of a multicast packet received the data, the transmission is complete and the multicast packet will be remove from the transmission queue.

2.4. Properties of SNMS

- There is neither hidden-terminal problem nor exposed-terminal problem could happen. Because SNMS uses the sender-based data transmission strategy and half-duplex operations, it is impossible that a node can receive data packets from two different senders at the same time. Thus, there is no hidden terminal problem could happen. The exposed-terminal problem is also resolved by the sender-based data transmission strategy.
- The data packets are collision free. Once a sender received non-zero bits during the reserve field of a frame, it will send the data packet on its channel and the receivers will receive the data packet on the same channel. Because these receivers can hear the signals only on sender's channel during the frame, and nodes in two-hop neighborhood use different channels, no collision of data packet will occur.
- SNMS avoids the waste of time slots. In our design, if none of the receivers can accept the data, the multicast frame will be terminated. This scheme avoids the waste of time slots.
- SNMS achieves higher reliable multicast efficiency. In SNDR, a packet can only received reliably by one receiver during a frame. On the other hand, in SNMS, a packet can be received reliably by several neighbors simultaneously.

The additional overhead $((3D+\log N)/8)$ bytes in a frame, where D is the maximum neighbors a node can have and N is the number of nodes in the network) for the SNMS transmitted in the ad hoc network can be ignored. Because SNMS can deliver the same packet to multiple neighbors at the same time, it is clear that the overall throughput in SNMS can be improved.

3. SIMULATION RESULTS AND DISCUSSION

3.1. Simulation parameters and performance metrics

We have implemented a simulation to evaluate the performance of our protocol. SNDR was used as a reference for comparison. Two hundred hosts were generated randomly in a physical area of size 100×100. The Maximum degree of the ad hoc network is 16, and the length of data packet is 1200 bits. The transmission range of every node is 10. Each channel's bandwidth is 1 Mbits/sec. The packet arrival rate, the network size and the length of data packets are varied in the simulation to investigate the impact of each of these parameters on the performance of SNDR and SNMS. Table 2 shows the default values and the range of each parameter. When one parameter is varied, other parameters are set to the default values shown in Table 2.

We use the throughput of the ad-hoc network to measure the effectiveness of SNMS. The throughput is defined in the following way:

$$\text{Throughput} = \frac{\text{Length of the Data Packet} * \text{Number of the Successful Packets}}{\text{Total Time}}$$

3.2. Impact of packet arrival rate

Figure 2 shows the throughput under the packet arrival rates from 80 to 800 packets/sec/host. In this figure, it is shown that the throughput increases as the packet arrival rate increases in SNMS and SNDR. At the lower packet arrival rates, the difference of throughputs between the two methods is small. However, SNMS gains the higher throughput in the higher packet arrival rates. This is because both SNMS and SNDR can deliver nearly all of the packets when the arrival rate is low. Both protocols (SNMS and SNDR) complete the light task in a relatively long time. Since SNDR transmits the same multicast packets several times but SNMS does only once. As the arrival rate (workload) increases, SNDR cannot complete as much work as SNMS can. SNMS provides higher throughput in comparison to the throughput obtained when employing SNDR.

3.3. Impact of the network size

In this experiment, we vary the network size from 50 to 400 nodes to observe the effect. Figure 3 shows the throughput of both protocols in this experiment. It is shown that the throughput increases as the network size increases in both SNMS and SNDR. Even if the throughput of SNMS is almost the same as that of SNDR while the network size is small, SNMS still obtains the higher throughput in the network of larger size. It is due to the following reason: With a smaller network size, the density is small (the area size is fixed) and so is the average degree of mobile nodes. The number of destinations of multicast is small so that SNMS cannot benefit from multicast. That is why both protocols have the same performance in the network of small size. As the network size increases, the workload of total network

TABLE 2
SIMULATION PARAMETERS

Parameter	Typical Value	Range
Arrival Rate	320 packets/sec/host	80 – 800 packets/sec/host
Network Size	200 nodes	50 – 400 nodes
Size of Data Packet	1200 bits	480 – 3120 bits

Fig. 2. Throughput of SNMS and SNDR with various arrival rates

Fig. 3. Throughput of SNMS and SNDR with various network sizes

increases too and so do the throughputs of both protocols. However, SNMS transmits to more destinations than SNDR do at the same time. As observed in Fig. 3, SNMS gains higher throughput.

3.4. Impact of the network size

Figure 4 indicates the impact of the length of data packets. As observed in the figure, the throughput of SNMS and SNDR increase as the size of data packet increases. The increasing of two protocols slow down while data packet size is larger than 1680 bits. It is due to the multicasting packet generated is reaching or over the load that both protocols can provide. Since SNDR adopts the receiver-based strategy, a sender in SNDR needs to wait for frame start signal issued

Fig. 4. Throughput of SNMS and SNDR with various lengths of data packet

from the receiver to send data on the receiver's channel. The sender idles during the period of waiting. In SNMS, which adopts the sender-based strategy, a sender sends the frame start signal on control channel and then some intended receivers join the transmission on the sender's channel. If all nodes in the receiver list are too busy to receive the multicast packet, the frame will end up after control field (Part C). The sender in SNMS does not need to wait for the receivers. The larger the size of data packet, the longer time that a sender should wait in SNDR. Therefore, the performance of SNMS is superior to that of SNDR.

4. CONCLUSION

In this paper, we have proposed a reliable MAC protocol, SNMS, to support multicast efficiently for multi-channel ad-hoc networks. In multi-channel environment, SNMS provides collision-free in the transmission of data packets by adopting neighbor-sequenced method. With the benefit of the sender-based strategy, SNMS delivers multicast packets to receivers with a smaller number of transmissions. Simulation results shown that SNMS achieves higher throughput than SNDR does in various arrival rates, network sizes and data packet lengths. Most importantly, the average throughput of SNMS can be improved up to 67% in comparison with SNDR. Accordingly, SNMS can support routing protocols achieving higher multicast efficiency.

REFERENCES

- [1] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A media access protocol for wireless LAN's," *Proc. ACM SIGCOMM*, pp. 212-225, 1994.
- [2] Z. Cai and M. Lu, "SNDR: a new medium access control for multi-channel ad hoc networks," *Proc. IEEE VTC*, vol. 2, pp. 966-971, 2000.
- [3] C. L. Fullmer and J. J. Garcia-Luna-Aceves, "Solution to hidden terminal problems in wireless networks," *Proc. ACM SIGCOMM*, 1997.
- [4] P. Karn, "MACA - a new channel access method for packet radio," *Proc. ARRL/CRRRL Amateur Radio 9th Comput. Networking Conf.*, 1990.
- [5] J. Jubin and J. D. Tornow, "The DRAPA packet radio network protocols," *Proceedings of the IEEE*, January 1987.

- [6] K. Tang and M. Gerla, "MAC reliable broadcast in ad hoc networks," *MILCOM 2001 Communications for Network-Centric Operations: Creating the Information Force*. IEEE, Vol. 2, pp. 1009-1013, 2001.
- [7] IEEE P802.11 Draft Standard for Wireless LAN: Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE, July 1996.