Reliable Multicast MAC Protocol for Wireless Ad Hoc Networks

Sung Won Kim[1] and Byung-Seo Kim[2]

[1] School of Electrical Engineering and Computer Science, Yeungnam University,
Gyeongsangbuk-do, 712-749, Korea
`ksw@ieee.org`
[2] Motorola Inc., 1301 Algonquin Rd. Schaumburg, IL, 60196 USA
`Byungseo.Kim@motorola.com`

Abstract. Reliable multicast in wireless ad hoc networks is important for the applications such as distributed computing, chat, and whiteboard. Due to the wireless channel characteristics, obtaining the reliability in multicast packet transmission is a difficult and challenging task. Even though IEEE 802.11 is widely adopted medium access control (MAC) protocol in wireless ad hoc networks, it does not support reliable multicast. Thus, several MAC layer protocols have been proposed that can provide reliable multicast. In this paper, we propose the reliable multicast MAC protocol which is based on the orthogonal frequency division multiple access (OFDMA) systems. The proposed method reduces the overhead required for control packets. Analysis results validate the reliability and efficiency of our multicast MAC protocol.

1 Introduction

Multicast or broadcast in the medium access control (MAC) layer refers to the process of sending data packets to some or all of the neighbors of a node. Thus, broadcast can be seen as a special case of multicast. Multicast communications are necessary for delivering acceptable quality of service in many applications of wireless communications such as distributed computing, chat, whiteboard, emergency report, and video conference. Several higher layer protocols rely heavily on reliable and efficient MAC layer multicast.

When data has to be sent to multiple recipients, multicast incurs less network cost compared to unicast. Multicast limits the transmission of redundant data and saves the bandwidth as well as energy. However, most research on wireless ad hoc networks have focused on reliable unicast. Reliable multicast has been studied relatively little compared to reliable unicast. In the IEEE 802.11 specification [1], the multicast sender simply listens to the channel and then transmits its data packet when the channel becomes idle for a period of time. There is no MAC-level recovery on multicast packet. As a result, the reliability of multicast is reduced as the probability of packet loss resulting from interference or collisions increases.

J. Indulska et al. (Eds.): UIC 2007, LNCS 4611, pp. 276–284, 2007.

Recently, a few multicast MAC protocols have been proposed to enhance the reliability and the efficiency of the IEEE 802.11 multicast protocol. Reliable multicast at the MAC layer proposed for IEEE 802.11 in the literature can be categorized into two basic types : multiple ACKs and leader-based ACK. In the multiple ACKs scheme, the sender collects the information of multicast packet reception from all of the multicast group member nodes. However, multiple ACK transmissions degrade the channel efficiency and reduce the overall network performance. The degradation is exaggerated as the number of member nodes increases. On the other hand, leader-based ACK scheme reduces the overhead caused by multiple ACK packet transmissions by allowing only a leader to send an ACK. Thus, the overhead of a leader-based ACK scheme is just the same as that of the unicast. If a node other than the leader experiences a failure of a data packet reception, it transmits a negative ACK (NACK) to make a collision of ACK from the leader. By using the NACK, this scheme guarantees some degree of the reliability. However, when one of the member nodes fails to receive a data packet because of hidden node problems, it can not send a NACK packet since it can not recognize that the received packet is multicast and it is the destination of the packet. Since the node does not send a NACK, no collision of ACK is experienced at the sender. Therefore, leader-based ACK scheme may not be reliable in terms of the detection of the failed reception.

In this paper, we propose a reliable multicast MAC protocol. The proposed method adopts the RTS/CTS handshaking to mitigate the hidden node problem. To reduce the overhead caused by control packets, we introduce the Orthogonal Frequency Division Multiple Access (OFDMA) method. The remainder of this paper is organized as follows. The next section presents the related work. In Section 3, the proposed method is described. In Section 4, we investigate the enhancement of the proposed method with some numerical results. Finally, the paper is concluded in Section 5.

2 Related Work

The paper in [2] proposes Broadcast Medium Window (BMW). In BMW, a sender exchanges RTS, CTS, DATA, and ACK packets with one of the member nodes, and then RTS and ACK packets are exchanged with all of the member nodes. These RTS and ACK packets are transmitted through contention based channel access. As an enhanced version of BMW, Batch Mode Multicast MAC (BMMM) protocol is proposed in [3]. The transaction of BMMM between the sender and member nodes is a sequence of multiple RTS/CTS exchanges, data packet transmission, and multiple Request ACK (RAK)/ACK exchanges. During this sequence, there is no contention-based channel access. Therefore, compared to BMW, BMMM reduces the overhead due to multiple contention periods of the access channel for transmitting RTS/ACK. However, there is still overhead of multiple control packets of RTS, CTS, RAK, and ACK. This overhead increases as the number of nodes increases.

In [4], the authors propose Leader-Based Protocol (LBP) for multicast to reduce the overhead caused by multiple CTSs and ACKs. A sender in LBP selects one node among the multicast group member nodes, called a leader. Then, only the leader responses with a CTS and an ACK corresponding to the RTS and the data packet. If a member node fails to receive a data packet, it sends a NACK packet at the end of the data packet and this NACK causes a collision with the ACK from the leader. If there is a collision after the data packet transmission, the sender recognizes that at least one node fails to receive the data packet. In that case, it sends the data packet again. Even though LBP reduces the overhead, it suffers from the hidden node problems. If there is a hidden node to one of multicast group member nodes, the receiver may not detect the transmission and may not respond with NACK. Thus, the leader cannot detect the failure of the packet reception.

Multicast aware MAC Protocol (MMP) is proposed in [5]. Unlike the aforementioned protocols, MMP does not use RTS and CTS handshaking, but uses data and ACK. After a data packet is transmitted, all of the member nodes transmit their ACK packets to the sender following the pre-assigned sequential order. If a receiver fails to receive a data packet, it sends a NACK packet in its pre-assigned location. In that case, the sender retransmits the data packet. Thus, the ACK period of MMP increases as the number of group member increases. Because MMP does not use RTS and CTS, it suffers from the hidden node problem.

In the previous work, the control packets are the tradeoff between the reliability and the bandwidth efficiency. For example, if we use RTS/CTS control packets, we can reduce the failure such as hidden node problems. However, this also reduces the system throughput.

3 Proposed Multicast MAC Protocol

3.1 OFDMA-Based Reliable Multicast

We propose new types of CTS and ACK packets, called OFDMA-based CTS (OCTS) and OFDMA-based ACK (OACK). We also propose a new multicast method that utilizes the OCTS and the OACK, called OFDMA-based Reliable Multicast (ORM). Fig. 1 shows an example of the OFDMA frequency band. Sub-channel data carriers are orthogonal with each other and they are used to transmit OFDM symbols. OCTS and OACK are control packets consisting of a preamble and OFDM symbols with a cyclic prefix [6]–[8].

Each member node has a unique pre-assigned sub-carrier location. The process of assigning a unique sub-carrier location is described in the following subsection. When a member node receives a RTS packet from the sender, it allocates an OFDM symbol into the pre-assigned sub-carrier of the CTS packet. The allocated symbol is one of the two BPSK symbols, 1 or -1. A successful reception of the RTS packet is indicated by a BPSK symbol 1 on the sub-carrier. On the contrary, a BPSK symbol -1 indicates a failed reception of the RTS packet. If a member node can not demodulate even the MAC header of the RTS packet, it will not

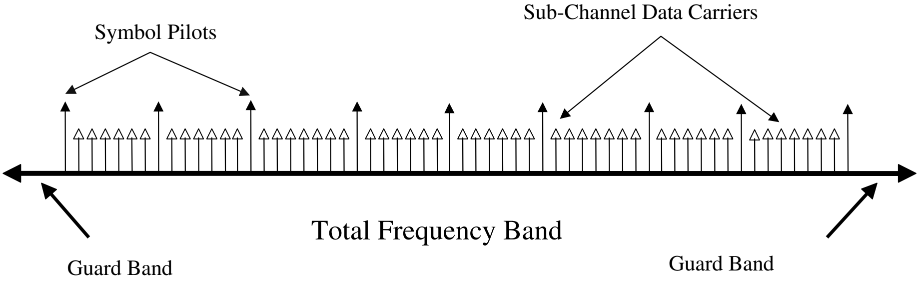

Fig. 1. An example of OFDMA sub-channel allocation

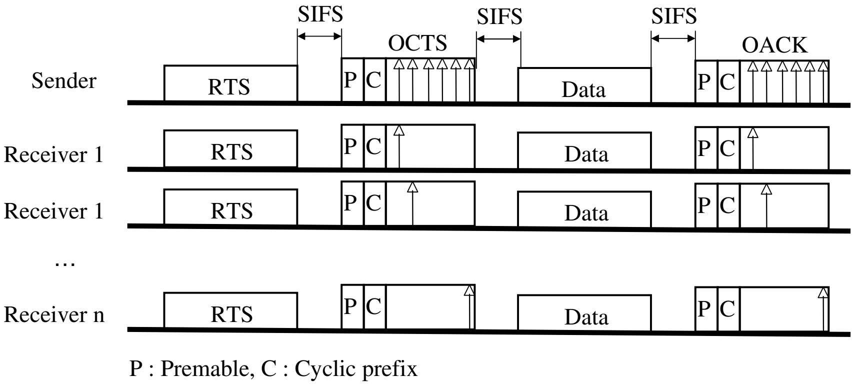

Fig. 2. An example of a data packet transmission cycle

generate an OFDMA symbol. An OFDM symbol generated by each member node for the acknowledgement has only one sub-carrier with a data symbol and the other sub-carriers are empty. The collection of these OFDMA symbols constitutes the payload of the OCTS packet. After being attached with the preamble, these OFDM symbols are sent to the multicast sender. It is assumed that all of the member nodes send their OFDMA symbols at the same time after SIFS idle period. At the multicast sender, the sub-carriers in the received OCTS are loaded by BPSK symbols to indicate each member's reception status.

An example scenario of the proposed method is shown in Fig. 2. A sender multicasts a RTS packet to the member nodes which are from Receiver 1 to Receiver n. All of the receivers response with an OFDMA symbol to the pre-assigned sub-carrier and these symbols are merged at the sender as an OCTS packet. Note that the payload part in OCTS packet is illustrated by frequency domain and the overall transmission sequence is illustrated by time scale in the figure. If Receiver n does not receive the RTS packet, it will not send an OFDMA symbol to the sender. After receiving OCTS, the sender checks the sub-carriers, which are assigned to member nodes. If any one of the member node sub-carriers is not allocated with any symbol or is allocated with the BPSK symbol -1, the sender prepares to retransmit the RTS packet.

When an OCTS is received correctly, the sender transmits a multicast packet to member nodes. When a member node receives a multicast packet from the sender, it allocates a symbol on the pre-assigned sub-carrier as an acknowledgement for the packet. The generation of OACK is the same as that of OCTS. An example scenario of OACK is illustrated in Fig 2.

For the time offset problem due to imperfect time-synchronization and different propagation delays from all of the member nodes, it is solved by using a longer cyclic prefix shown in [6]–[8] which is longer than a delay spread profiles.

3.2 Sub-channel Assignment

The sub-channel assignment is managed by a multicast leader (ML). Each multicast group has an ML. When a node wants to join a multicast group, it broadcasts a multicast join request (MJREQ) packet. When the ML receives the MJREQ, it assigns an empty sub-channel to the requesting node. Then, the ML responds with a multicast join acknowledgement (MJACK) packet that has the information of the allocated sub-channel. The assigned sub-carrier has to be unique for each node within the same multicast group address.

If there is no ML, MJACK is not responded. In that case, i.e., if there is no response within some time threshold, the requesting node becomes a new ML for that multicast group address.

When an ML wants to leave a multicast group, it unicasts a multicast leader request (MLREQ) packet to one of the multicast group members. If the node responds with a multicast leader acknowledgement (MLACK), the responding node becomes a new ML. If there is no MLACK within some time threshold, ML selects another node and transmits the MLREQ packet until the new ML is selected.

4 Packet Delay Analysis

4.1 Average Packet Delay

To compare the performance of the proposed method with that of MMP, we adopt the analytical model used in [9]–[13]. The channel conditions are assumed ideal, i.e., no hidden terminals, no channel error, and no capture effect. We consider a system consisting of N nodes. Each node always has a packet available for transmission. In other words, we operate in saturation conditions where the transmission queue of each node is assumed to be always nonempty. We consider a multicast group which has R member nodes. All the nodes are located in the transmission range of the sender.

The duration of the backoff is determined by the contention window (W) size which is initially set to W_{min}. The W value is used to randomly choose the number of slot time (σ) in the range of [0, $W-1$], which is used for backoff duration. In case of an unsuccessful transmission, the W value is updated to $2 \times W$ while it does not exceed W_{max}. Let us adopt the notation $W_i = 2 \times W_{i-1}$ where $i \in \{1, \ldots, m\}$ is called the backoff stage and m is the maximum backoff stage such that $W_{max} = 2^m W_{min}$.

We assume that all the nodes operate synchronously. A discrete and integer time scale is adopted: t and $t+1$ correspond to the beginnings of two consecutive changes of backoff time counter. We call the time interval between t and $t+1$ as "counter time slot". Note that the counter time slot (variable duration) is different with the slot time (constant duration). Since the decrement of backoff time counter is stopped when the channel is sensed busy, the time interval between the beginnings of two consecutive backoff time counter instants may be much longer than the constant slot time size .

Let us denote the event that a node transmits a packet into a counter time slot as X. We are interested in the *unconditional probability* $\tau = \text{Prob}(X)$ that a node transmits a packet into a counter time slot. Let p_c be the *conditional probability* that a transmitted packet sees a collision on the channel. Once the independence is assumed and p_c is supposed to be a constant value, it is possible to solve τ and p_c using numerical techniques. The numerical method for finding τ and p_c is illustrated in [10][13].

Let M be the number of counter time slots required for multicast receivers to receive the multicast packet successfully. Let $E[c_i]$ be the average value of the backoff counter extracted by a node entering stage i. $E[c_i]$ is equal to $W_i/2$ in the assumption of the uniform distribution in the range of $(0,\ W_i - 1)$. The average value of M is given as

$$E[M] = \sum_{i=0}^{m} \Big\{ 1 + E[c_i] \Big\} \pi_i, \tag{1}$$

where π_i is the steady state probability of backoff stage i [13].

Let T_{RTS}, T_{CTS}, T_{DAT}, T_{ACK}, T_{SIFS}, and T_{DIFS} be the transmission durations of RTS packet, CTS packet, data packet, ACK packet, SIFS, and DIFS, respectively. Let T_{tx} be the time duration of a data packet transmission cycle. T_{tx} includes all time durations required for a data packet transmission such as control packets, data packet, and IFS (refer to Fig. 2). In our proposed method, the average value of T_{tx} is given as

$$E[T_{tx}^{ORM}] = T_{RTS} + T_{CTS} + T_{DAT} + T_{ACK} + 3T_{SIFS} + T_{DIFS}, \tag{2}$$

because the new packet transmission starts after the DIFS duration. On the contrary, the average value of T_{tx} in MMP is given as

$$\begin{aligned} E[T_{tx}^{MMP}] = T_{RTS} + T_{CTS} + T_{DAT} + R \times (T_{ACK} + T_{SIFS}) \\ +2T_{SIFS} + T_{DIFS}, \end{aligned} \tag{3}$$

because MMP requires R ACK packets.

The mean value of counter time slot T_{ct} is calculated as follows. The probability that a sender sees an idle channel during a counter time slot is Prob[idle] = $(1-\tau)^N$, which takes time σ. The probability of successful transmission during a counter time slot is Prob[success] $= N\tau(1-\tau)^{N-1}$, which takes time T_{tx}. The probability of unsuccessful transmission during a counter time slot is Prob[fail] = 1 - Prob[idle]- Prob[success], which takes time T_{tx}. In ORM, the mean value of counter time slot, $E[T_{ct}^{ORM}]$, is given as

$$\begin{aligned} E[T_{ct}^{ORM}] &= \text{Prob[idle]}\sigma + \text{Prob[success]}E[T_{tx}^{ORM}] + \text{Prob[fail]}E[T_{tx}^{ORM}] \\ &= (1-\tau)^N\sigma + N\tau(1-\tau)^{N-1}E[T_{tx}^{ORM}] \\ &\quad +[1-(1-\tau)^N - N\tau(1-\tau)^{N-1}]E[T_{tx}^{ORM}] \\ &= (1-\tau)^N\sigma + [1-(1-\tau)^N]E[T_{tx}^{ORM}]. \end{aligned} \tag{4}$$

Similarly, the mean value of counter time slot of MMP is given as

$$\begin{aligned} E[T_{ct}^{MMP}] &= \text{Prob[idle]}\sigma + \text{Prob[success]}E[T_{tx}^{MMP}] + \text{Prob[fail]}E[T_{tx}^{MMP}] \\ &= (1-\tau)^N\sigma + N\tau(1-\tau)^{N-1}E[T_{tx}^{MMP}] \\ &\quad +[1-(1-\tau)^N - N\tau(1-\tau)^{N-1}]E[T_{tx}^{MMP}] \\ &= (1-\tau)^N\sigma + [1-(1-\tau)^N]E[T_{tx}^{MMP}]. \end{aligned} \tag{5}$$

The packet delay is defined as the time period from the start of a packet becoming a head-of-line (HOL) in the queue to the end of the packet removal from the queue [14]. The packet removal is caused by a successful reception by all the members. The sender must process every ACK received for the packet. Whenever the sender has not received ACKs from all the receivers for the packet, the packet must be re-multicasted, the backoff stage is increased, and the backoff timer is restarted. Let us denote the packet delay of ORM by D^{ORM}. Considering that the sender contends the channel for M counter time slots before the packet removal from the queue, average packet delay of ORM is

$$E[D^{ORM}] = E[M]E[T_{ct}^{ORM}]. \tag{6}$$

Similarly, the average packet delay of MMP is

$$E[D^{MMP}] = E[M]E[T_{ct}^{MMP}]. \tag{7}$$

From the second term of (5), it is noted that the effect of $E[T_{tx}^{MMP}]$ on $E[T_{ct}^{MMP}]$ increases as the number of nodes N increases. Also note in (3), (5), and (7) that the packet delay increases as the number of receivers R increases. Thus, the packet delay of MMP largely depends on the number of nodes. These trends will be shown in the next subsection.

4.2 Numerical Results

The values shown in the following figure have been obtained by using the system parameters in Table 1 and are based on the orthogonal frequency division multiplexing (OFDM) physical layer used in IEEE 802.11a standard [15].

Table 1. Parameter values

Parameter	Value
m	6
W_{min}	16
W_{max}	1024
SIFS time	16 μs
DIFS time	34 μs
slot time	9 μs
MAC header	272 bits
PHY header	46 bits
Preamble	16 μs
ACK time	44 μs
RTS time	52 μs
CTS time	44 μs
packet payload	8192 bits
channel bit rate	6 Mbps

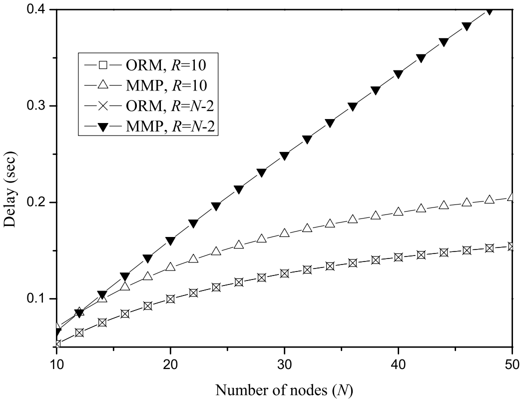

Fig. 3. Packet delay of ORM and MMP

Fig. 3 shows the packet delay of ORM and MMP. The number of receivers R is set to constant 10 and variable $N-2$. As the number of nodes increases, the delay also increases. This is because the average number of counter time slots $E[M]$ in (1) and the average value of counter time slot $E[T_{ct}]$ in (4) and (5) increases which affects the average packet delay $E[D]$ in (6) and (7), respectively. For MMP, the delay of variable R is larger than that of constant R. This is because the number of ACK packets increases according to the number of receivers. On the contrary, the delay of ORM is the same for the two cases of R. This is because ORM requires only one ACK packet irrespective of R. Thus the proposed method can show better performance than MMP when there are many multicast receivers.

5 Conclusion

We proposed a reliable multicast MAC protocol for wireless ad hoc networks. The proposed method adopts the RTS/CTS handshaking to mitigate the hidden node problem. The proposed method uses the OFDMA method in order to combine the multiple CTS and ACK packets. By reducing the time used for the CTS and ACK packets, the proposed method can reduce the packet transmission delay, significantly. The proposed method is robust to the increase of the number of receivers because the CTS and ACT time durations are irrespective of the number of receivers.

References

1. IEEE Std 802.11: 1999(E): Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (1999)
2. Tang, K., Garcia, M.: MAC reliable broadcast in ad hoc networks. In: Proc. IEEE MILCOM 2001, pp. 1008–1013 (2001)
3. Sum, M.T., Huang, L., Arora, A., lai, T.H.: Reliable MAC layer multicast in IEEE 802.11 wireless networks. Wireless Communication and Mobile Computing 3, 439–453 (2003)
4. Kuri, J., Kasera, S.K.: Reliable multicast in multi-access wireless LANs. ACM Wireless Networks 7, 359–369 (2001)
5. Gossain, H., Nandiraju, N., Anand, K., Agrawal, D.P.: Supporting MAC layer multicast in IEEE 802.11 based MANETs: Issues and solutions. In: Proc. IEEE LCN 2004, pp. 172–179. IEEE Computer Society Press, Los Alamitos (2004)
6. Cao, Z., Tureli, U., Yao, Y.D.: Deterministic multiuser carrier-frequency offset estimation for interleaved ofdma uplink. IEEE Trans. Commun. 52, 1585–1594 (2004)
7. Kaiser, S., Krzymien, W.A.: Performance effects of the uplink asynchronism in a spread spectrum multicarrier multiple access system. Eur. Trans. Commun. 10, 399–406 (1999)
8. Kapoor, S., Marchok, D.J., Huang, Y.F.: Adaptive interference suppression in multiuser wireless ofdm system using antenna arrays. IEEE Trans. Signal Processing 47, 3381–3391 (1999)
9. Bao, C.W., Liao, W.: Performance analysis of reliable MAC-layer multicast for IEEE 802.11 wireless LANs. In: Proc. IEEE ICC 2005, pp. 1378–1382 (2005)
10. Bianchi, G.: Performance analysis of the IEEE 802.11 distributed coordination function. IEEE J. Select. Areas Commun. 18, 535–547 (2000)
11. Vitsas, V.: Throughput analysis of linear backoff scheme in wireless LANs. IEE Elect. Lett. 39, 99–100 (2003)
12. Hui, J., Devetsikiotis, M.: Designing improved MAC packet schedulers for 802.11e WLAN. In: Proc. IEEE Globecom 2003, San Francisco, CA, USA, IEEE Computer Society Press, Los Alamitos (2003)
13. Kim, S.W., Kim, B., Fang, Y.: Downlink and uplink resource allocation in IEEE 802.11 wireless LANs. IEEE Trans. Veh. Technol 54, 320–327 (2005)
14. Towsley, D., kurose, J., Pingali, S.: A comparison of sender-initiated and receiver-initiated reliable multicast protocols. IEEE J. Select. Areas Commun. 15, 398–406 (1997)
15. IEEE Std 802.11a-1999: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-speed Physical Layer in the 5 GHz Band (1999)