International Journal of Engineering Research and Technology.
ISSN 0974-3154 Volume 11, Number 8 (2018), pp. 1285-1300

http://www.irphouse.com

Collision Free Hybrid Slot Protocol for Improving Performance in Wireless Networks

Sunmyeng Kim

Department of Computer Software Engineering,
Kumoh National Institute of Technology, South Korea.

Abstract

To support quality of service (QoS), IEEE 802.11e, IEEE 802.11aa, and IEEE 802.11p standards use the enhanced distributed channel access (EDCA) mechanism. The EDCA provides QoS by introducing four access categories (ACs). As the number of stations increases, performance of the EDCA is getting worse because it is a contention-based channel access mechanism. For improving the performance of the EDCA mechanism, an efficient collision resolution protocol must be used. To this end, many collision reduction protocols have been proposed. However, there remain still many problems. In this paper, we propose a novel hybrid slot protocol to resolve collisions and to avoid the hidden terminal problem. The proposed protocol uses short-duration busy-tone signals and defines new control packets. Stations are synchronized by broadcasting busy tones to neighbours. Therefore, the hidden terminal problem can be avoided. The proposed protocol is based on hybrid slot which consists of several mini time slots. The mini time slots are allocated to each AC through the exchange of new control packets. And only one AC transmits its data packet in a mini time slot. Consequently, the proposed protocol can avoid data packet collisions completely. Simulation results show that the proposed protocol works well and improves performance.

Keywords – Collision, EDCA, Hidden Terminal, Hybrid Slot, MAC.

I. INTRODUCTION

The IEEE 802.11 wireless LAN is widely used for wireless access due to its easy deployment and low cost. The IEEE 802.11 standard defines a medium access control (MAC) protocol for sharing the channel among nodes [1]. The distributed coordination function (DCF) was designed for a contention-based channel access. The

DCF has two data transmission methods: the default basic access and optional RTS/CTS (request-to-send/clear-to-send) access. The basic access method uses the two-way handshaking (DATA-ACK) mechanism. The RTS/CTS access method uses the four-way handshaking (RTS-CTS-DATA-ACK) mechanism to reserve the channel before transmitting long data packets. This technique is introduced to avoid the hidden terminal problem.

The widespread use of multimedia applications requires new features such as high bandwidth and small average delay in wireless LANs. Unfortunately, the IEEE 802.11 MAC protocol cannot support quality of service (QoS) requirements [2, 3]. In order to support multimedia applications with tight QoS requirements in the IEEE 802.11 MAC protocol, the IEEE 802.11e has been standardized [4]. It introduces a contention-based new channel access mechanism called enhanced distributed channel access (EDCA). The EDCA supports the QoS by introducing four access categories (ACs). To differentiate the ACs, the EDCA uses a set of AC specific parameters, which include minimum contention window (CWmin), maximum contention window (CWmax), and arbitration inter-frame space (AIFS). The EDCA also introduces a TXOP (Transmission Opportunity) parameter to provide service differentiation and QoS of the traffic.

The IEEE 802.11aa standard extends the IEEE 802.11e standard to increase the granularity of traffic prioritization and QoS support provided by the EDCA [5, 6]. The 802.11aa standard further differentiates the priority of video and voice streams. To achieve this, it divides the transmission queues for video AC and voice AC into a primary queue and an alternative queue.

IEEE 802.11p is an amendment to the IEEE 802.11 standard to support wireless access in vehicular environments (WAVE) [7] or vehicular ad hoc networks (VANETs). It uses the EDCA mechanism to support different levels of QoS.

IEEE 802.11e, IEEE 802.11aa, and IEEE 802.11p use the similar EDCA mechanism to provide QoS. Performance analysis of the EDCA mechanism has recently been studied in the literature [8-15]. We can summarize the analysis as follows. Though EDCA mechanism can provide differentiated services, there is still a probability that nodes select the same backoff value, and collisions occur among data packets. The EDCA mechanism can be inefficient for providing QoS because of high collision probability in networks with many stations.

Collisions are unavoidable since the EDCA is a contention-based access mechanism. The EDCA incurs two types of collisions: virtual and physical collisions. The virtual collisions involve two or more AC queues belonging to the same station. When the backoff counters of more than one queues in the same station reach zero and try to transmit their data packets at the same time, it leads to a virtual collision. To handle the virtual collision, the virtual collision handler specified in the 802.11e standard grants a transmission opportunity to the AC with higher priority, whereas the AC with lower priority starts new backoff process. The real collisions can be avoided by this handler. The physical collisions involve queues from two or more different stations. When the backoff counters of several contending queues in different stations expire at the same time, they simultaneously transmit their data packets on the channel. Therefore, they make a physical collision one another. The physical collisions can be classified into inter-AC and intra-AC collisions. The inter-AC collisions occur among data packets of different AC queues in two or more different stations when the backoff counters for the different AC queues expire simultaneously. The intra-AC collisions occur among data packets of the same AC queues in two or more different stations. Hereafter, collision means a physical collision.

In order to improve the performance of the EDCA mechanism, an efficient collision resolution protocol must be used. To this end, many collision reduction protocols have been proposed [8-10, 16, 17]. Khatua et al. [8] proposed a delay-aware distributed dynamic adaptation of contention window scheme, namely D2D, for the cumulative improvement of both the throughput and the channel access delay at runtime. The D2D mechanism requires two estimates - delay deviation ratio and channel busyness ratio - on present delay level and congestion status, respectively. The D2D performs a probabilistic resetting of CW to its minimum value based on the estimated delay deviation ratio and channel busyness ratio after a frame transmission. Gao et al. [9] proposed an adaptive tuning (AT) scheme of the backoff parameters such as initial backoff window sizes and AIFS numbers to optimize the network throughput. The AP keeps track of the number of nodes of each AC accordingly, and calculates the optimal backoff parameters. And then it broadcasts the updated backoff parameters in each beacon frame. After receiving the beacon frame, each node sets its own backoff parameter, and then operates same as the EDCA mechanism. Chang et al. [10] proposed an earliest deadline first based carrier sense multiple access (EDF-CSMA) scheme. The EDF-CSMA dynamically adjusts the priority of real-time streaming to avoid collision. EDF-CSMA provides a collision-free QoS for nodes by introducing the concept of EDF scheduling and reduces controlled packet transmission by integrating the properties of CSMA/CA distributed access. Yao, et al. [16] proposed a hybrid slot allocation mechanism (HSAM) to reduce the collisions among different ACs. In the HSAM, three time slots are assembled into a hybrid slot, each slot in the hybrid slot is allocated to a particular AC according to its priority. If the backoff counters of one or more ACs expire at the same time, then they transmit their data packets in the slot allocated to each AC.

Although many protocols have been proposed, there remain still many problems. The D2D protocol is based on measurement. Therefore, it is hard to make a right decision since the protocol cannot accurately measure the network situation at runtime. The AT protocol is based on analytical model. The analytical model is usually derived based on a few unpractical hypotheses. They do not reflect the characteristics of real situation. Therefore, the model-based mechanism is always inaccurate and clearly not applicable to realistic environments. The EDF-CSMA is only applicable for single video AC. It did not propose any mechanism for the other ACs. The D2D, AT, and

EDF-CSMA protocols do not avoid the collisions even though they reduce the collisions. The HSAM protocol does not avoid the collisions and can waste channel time if there are no ACs of which backoff counters reach zero during the hybrid slot. Also it can cause the hidden terminal problem.

In this paper, we propose a novel collision free hybrid slot (CFHS) protocol to resolve inter-AC and intra-AC collisions and hidden terminal problem. The proposed CFHS protocol has the similar concept of hybrid slot to that of the HSAM protocol, and uses short-duration busy-tone signals (i.e., pulses of energy) [18].

The paper is organized as follows. In Section II, we give a brief introduction of the IEEE 802.11e EDCA and the HSAM protocol, which is one of typical collision resolution MAC protocols. In Section III, the proposed CFHS protocol is presented in detail. In Section IV, performance studies are carried out through simulation results. Finally, we draw a conclusion in Section V.

II. RELATED WORK

In this Section, we summarize the HSAM protocol proposed in [16].

The HSAM protocol was proposed to reduce the collision probability and improve the transmission performance of each AC while guaranteeing service differentiation simultaneously. The HSAM protocol is based on hybrid slot which consists of three time slots. The first time slot is allocated to the highest priority AC 3, the second one is allocated to the medium priority AC 2, and the last one is shared by AC 1 and AC 0 due to their lower priorities. Each AC in stations performs its own backoff procedure. When the backoff counter of an AC reaches zero, it transmits its data packet during the time slot allocated to the corresponding AC.

Fig. 1 shows an example of the HSAM protocol. In the figure, two stations (A and B) exist. Each AC in them performs its own backoff procedure to transmit data packets. The backoff counters of AC 3 and AC 2 in station A, and AC 2 in station B become zero at the same time. AC 3 in station A transmits its own data packet during the first time slot. After successfully receiving the packet, the AP sends an ACK packet. AC 2 in station A and AC 2 in station B simultaneously transmit their data packets during the second time slot. And they make a collision. Therefore, the AP does not send an ACK packet. No one transmits a data packet during the third time slot because there are no AC 1 and AC 0 of which backoff counters reach zero.

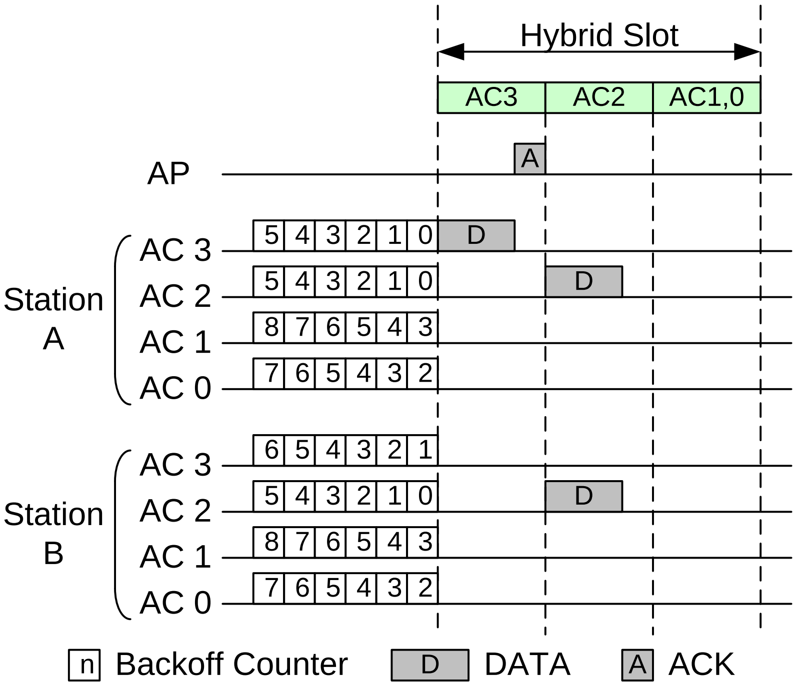

Fig. 1 Example of HSAM protocol

The HSAM protocol still has four problems. 1) The HSAM protocol always allocates time slots for ACs in a hybrid slot. If there are no ACs of which backoff counters reach zero, they do not transmit their data packets, cause empty slots and waste time slots (e.g., the third time slot for AC 1,0 in Fig. 1). 2) In the HSAM protocol, intra-AC collision cannot be avoided because two or more stations can simultaneously transmit their data packets during one time slot (e.g., the second time slot for AC 2 in Fig. 1). 3) The HSAM protocol does not avoid inter-AC collisions due to the lack of synchronization. The HSAM protocol just allows every station to start the hybrid slot whenever they have the backoff counter reaching zero. When stations transmit data packets without synchronizing with one another, there is always the possibility of the collision of data packets. Fig. 2 shows an example of two stations A and B which are not synchronized. AC 2 and AC 1 in station B have backoff counter of zero at time t0, and station B start the hybrid slot at t0. Station A does not know the beginning of hybrid slot at station B. Therefore, station A continues its backoff procedure. AC 3 and AC 2 in station A have backoff counter of zero at time t1, and station A starts the hybrid slot at t1. AC 3 in station A transmits its data packet at t1, and AC 2 in station B transmits its data packet at t2. They make a collision because their data packet are overlapped. 4) The HSAM protocol does not consider hidden environments. Therefore, it can cause the hidden terminal problem.

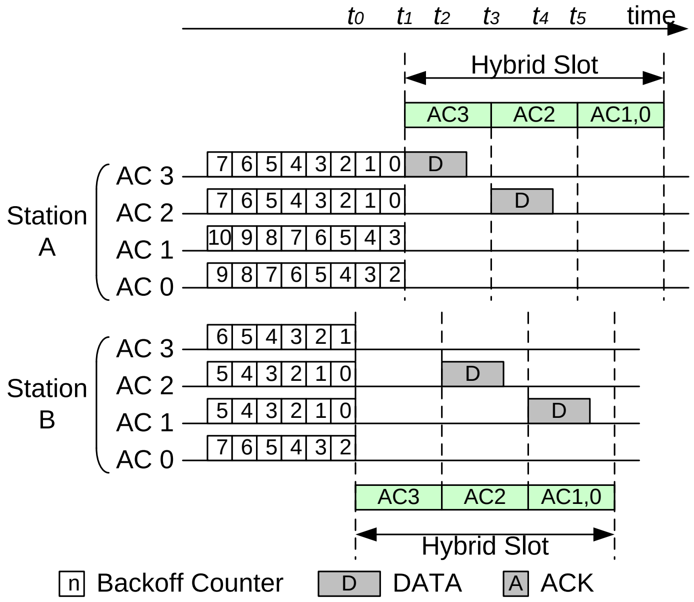

Fig. 2 Synchronization problem of HSAM protocol

III. PROPOSED CFHS PROTOCOL

In this Section, we describe a proposed CFHS protocol to solve the problems of the HSAM protocol and improves the network performance. The proposed CFHS protocol has the similar concept of hybrid slot to that of the HSAM protocol.

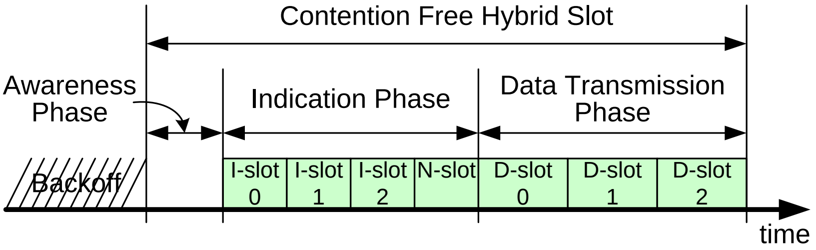

Fig. 3 Structure of CFHS protocol

The proposed CFHS protocol consists of 3 phases: awareness phase, indication phase, and data transmission phase (see Fig. 3). In the awareness phase, every station in the network is synchronized and is aware of the start of indication phase and data transmission phase. In the indication phase, there are three indication time slots (I-slots 0, 1, and 2) and one notification time slot (N-slot). I-slots 0, 1, and 2 are used by AC 3, AC 2, and AC 1/0 to send control packets for indicating their backoff counters expire, respectively. N-slot is used by AP to notify the ACs of the result of the indications (success or failure). The data transmission phase has three data time slots (D-slots 0, 1, and 2). D-slots are used by ACs of which indication is successful to transmit their data packets.

Before describing the operation of each phase in detail, we explain new three control packets: IND (Indication), NOTI (Notification), and DT-End + ACK (Data Transmission phase-End + ACK). Their formats are defined in Fig. 4.

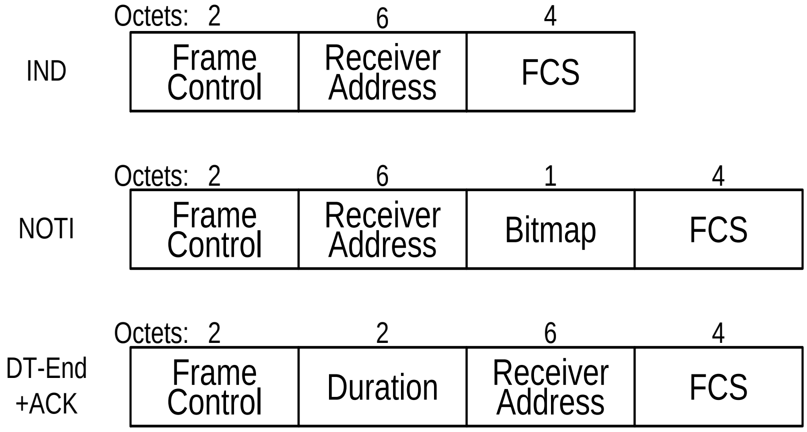

Fig. 4 Control packet formats

IND packet is used by stations during I-slots to indicate that they have ACs of which backoff counters expire. NOTI packet is used by AP during N-slot to notify stations of the result of the indications. The Receiver Address field is the broadcast address. NOTI packet includes a bitmap. Each bit in the bitmap shows the status of indications. ‘1’ means the success of an indication, while ‘0’ means the failure. DT-End + ACK packet is used to indicate the end of a data transmission phase and acknowledge receipt of a data packet without error. The duration field is set to 0. Therefore, NAV (Network Allocation Vector) value of each station is initialized when it receives a DT-End + ACK packet.

Hereafter, we explain the operation of CFHS protocol in detail. In the proposed CFHS protocol, each AC starts the awareness phase after its backoff counter expires (see Fig. 3). Henceforth, for the purpose of classification, we use terms Exp AC and Non-exp AC. Exp AC is an AC of which backoff counter expires, and Non-exp AC is an AC of which backoff counter does not expire.

In the awareness phase, the proposed CFHS protocol uses two busy tones. An Exp AC first transmits a busy tone to indicate that its backoff counter expires. On receiving the busy tones from Exp ACs, AP sends an another busy tone at the next slot. Therefore, every AC in a network can recognize the presence of the Exp ACs and the beginning of the indication phase. Exp ACs enter the indication phase after receiving the busy tone from AP. Non-exp ACs defer their backoff procedure until the end of the data transmission phase. In the proposed CFHS protocol, stations are synchronized by broadcasting busy tones to neighbours during the awareness phase. Therefore, the CFHS protocol can avoid the inter-AC collisions.

By using two busy tones, all ACs are aware of the start of indication phase even in hidden terminal environments. When Non-exp ACs receive the busy tones from Exp ACs and/or AP, they set the NAV and do not contend for the channel access in order to avoid collisions. The NAV is set as followings;

$$NAV = ID\_Phase + DT\_Phase \tag{1}$$

$$ID\_Phase = n \times I\_slot + N\_slot \tag{2}$$

$$DT\_Phase = n \times D\_slot \tag{3}$$

where, *ID_Phase* and *DT_Phase* are the time durations for indication and data transmission phases, respectively. *I_slot* and *N_slot* are the time durations of an I-slot and an N-slot, respectively. *D_slot* is the time duration to transmit maximum DATA packet and ACK. n is the maximum number of I-slots and D-slots.

Kang et al. [19] explored how the number of slots in a multi-slot protocol influences the performance. Based on the numerical analysis, they concluded that three slots are appropriate in the multi-slot protocol. This number has been adopted in the proposed CFHS protocol. Therefore, each phase consists of three time slots (n=3). The proposed CFHS protocol works well even though any number of time slots is selected.

After receiving the busy tone from the AP, Exp ACs start the indication phase (see Fig. 3). Exp ACs send IND packets in corresponding I-slots. After receiving IND packets from the Exp ACs, AP checks the I-slots in which it successfully receives IND packets. If no Exp ACs send IND packets, AP does not receive any IND packet. If only one Exp AC sends an IND packet, AP successfully receives the IND packet. And if two or more Exp ACs send IND packets, AP receives erroneous IND packets. Receipt of non IND packets or erroneous IND packets means the indication failure, while receipt of one IND packet means the indication success. And then, AP transmits an NOTI packet in N-slot to notify the Exp ACs of the result of the indications. NOTI packet includes a bitmap of 1 byte. In the bitmap, ‘1’ means the indication success, while ‘0’ means the failure. There is a one-to-one mapping from bits in the bitmap to I-slots, i.e., bitmap[i] corresponds to the I-slot i.

After receiving the NOTI packet from AP, Exp ACs check whether their indications are successful or not. They can know easily the result of the indications by looking into the bitmap in the NOTI packet. Henceforth, for the purpose of classification, we use terms Succ AC and Fail AC. Succ AC is an AC of which indication succeeds, and Fail AC is an AC of which indication fails. The Succ ACs can determine their D-slots in the data transmission phase for DATA/ACK as follows. The first bit position which is set corresponds to the Succ AC that has to transmit a data packet in the first D-slot. The second bit position that is set corresponds to the Succ AC that should occupy the second D-slot and so on. Specifically, the Succ AC that sent an IND packet in the I-slot *i* occupies D-slot *j* if bitmap[*i*] is the *j*th bit that is set.

Non-exp and Fail ACs set their NAV values not to contend for the channel access. The NAV value is set as followings;

$$NAV = k \times D\_slot \tag{4}$$

where, k is the number of bits which are set to 1 in the bitmap field of the NOTI packet.

After determining their D-slots, Succ ACs enter the data transmission phase. Each Succ AC transmits its own data packet in its D-slot. AP receives a data packet in a D-slot and transmits an ACK packet to the Succ AC. In the last D-slot, AP transmits a DT-End+ACK packet instead of ACK packet to notify all the ACs of the end of data transmission phase. After receiving the DT-End+ACK packet from AP, all the ACs reset their NAV values and contend for the channel access again.

The proposed CFHS protocol does not waste channel time because AP only allocates D-slots for the ACs which successfully sent IND packets. Also the CFHS protocol avoids the intra-AC collisions because only one Succ AC transmits a data packet in a D-slot.

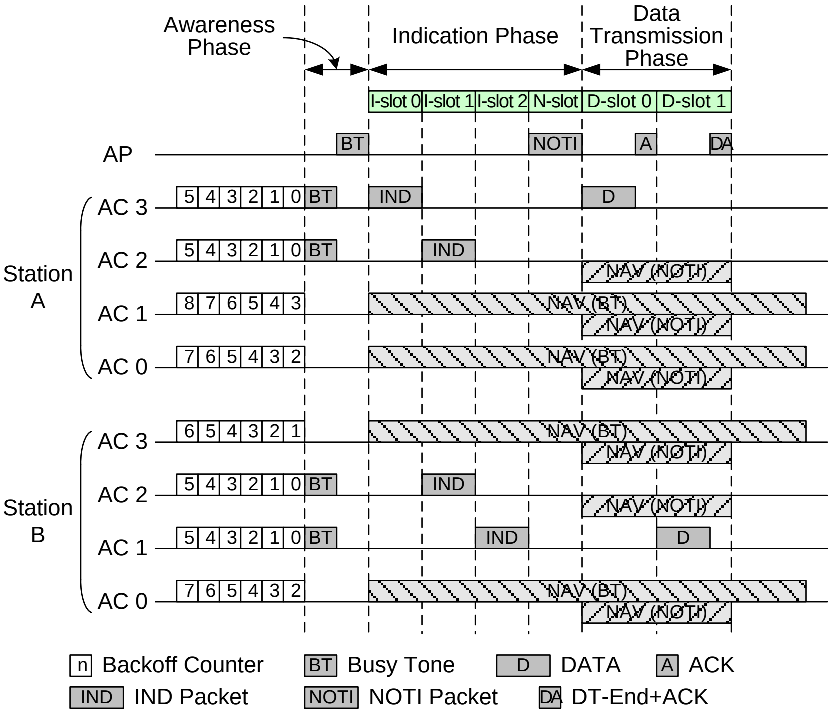

Fig. 5 Operation of the proposed CFHS protocol

Fig. 5 presents an example of the operation of the proposed CFHS protocol. In the figure, two stations A and B exist. Each station has four ACs (AC 3, AC 2, AC 1, and AC 0). Each AC in the stations performs its own backoff procedure to transmit data packets. The backoff counters of AC 3 and AC 2 in station A, and AC 2 and AC 1 in station B expire at the same time. They become Exp ACs, and the others become Non-exp ACs. Exp ACs enter the awareness phase and transmit busy tones. The AP sends another busy tone at the next time slot. After receiving the busy tone from AP, Non-exp ACs set their NAV values by using equations (1), (2), and (3) not to contend for the channel access. Exp ACs start the indication phase after receiving the busy tone from AP. Exp AC 3 in station A transmits an IND packet in I-slot 0. Exp AC 2s in station A and B simultaneously transmit IND packets in I-slot 1. Therefore, they make collisions. Exp AC 1 in station B transmits an IND packet in I-slot 2. The AP receives the IND packets in I-slots 0 and 2 without collisions. However, it receives collided IND packets in I-slot 1. Therefore, it sets bits 0 and 2 in a bitmap to 1. The other bits are set to 0. The bitmap is 10100000. Then, AP transmits an NOTI packet including the bitmap. After receiving the NOTI packet from AP, each Exp AC checks the bitmap to see whether its indication is successful or not. Exp AC 3 in station A and Exp AC 1 in station B become Succ ACs because bits 0 and 2 in the bitmap are set to 1. Recall that they sent IND packets in I-slots 0 and 2, respectively. The other Exp ACs become Fail ACs. Non-exp ACs and Fail ACs set their NAV values based on equation (4). The Succ ACs determine their D-slots. D-slots for the Succ AC 3 in station A and Succ AC 1 in station B are 0 and 1, respectively.

After determining their D-slots, the Succ ACs enter the data transmission phase. The Succ AC 3 in station A transmits its own data packet in D-slot 0 and AP transmits an ACK packet. The Succ AC 1 in station B transmits its own data packet in D-slot 1 and AP transmits a DT-End+ACK packet because D-slot 1 is the last slot in the data transmission phase. After receiving the DT-End+ACK packet from AP, all the ACs reset their NAV values (these are not drawn on Fig. 5) and contend for the channel access again.

IV. SIMULATION RESULTS

Let us discuss the simulation results of the proposed CFHS protocol. To validate the proposed protocol, we compare them to the results of the HSAM protocol. In the simulation, a network with one AP and n stations is considered. System parameters used in the simulation are listed in Table 1. We simulated an IEEE 802.11a network with transmission rates of 54 Mbps for data packets and of 6 Mbps for control packets, respectively. In the simulation, we consider the basic access method and only uplink traffic. Each station has three types of AC (AC 3, AC 2, and AC 1). AC parameters are listed in Table 2. A constant data packet size of 1,000 bytes was used. Simulations run for 100 seconds, and all simulation results are averaged over ten simulations.

Table 1. Simulation parameters

Parameter	Value
Data Rate	54 Mbps
Control Rate	6 Mbps
Slot Time	9 us
SIFS	16 us
Retry Limit	7
Propagation Delay	1 us
MAC Header	26 Bytes
FCS	4 Bytes
ACK	14 Bytes
PHY PLCP Preamble Length	16 us
PHY PLCP Header Length	5 Bytes

Table 2. AC parameters

Parameter	AC 3	AC 2	AC 1
AIFSN	2	4	7
CWmin	7	15	31
CWmax	15	31	1023
Data Size (Bytes)	1000	1000	1000

We conducted simulations under saturated and unsaturated traffic environments. In the saturated environments, each AC always has data packets to transmit, and all packets are of the same size. In the unsaturated environments, we use the negative exponential distribution to get the lengths of the data packet inter-arrival times. The average inter-arrival time of the distribution with arrival rate parameter λ is $1/\lambda$. In the simulation, the arrival rate is set to 400. It means that each AC generates 400 data packets per second on average (i.e., a rate of 3.2 Mbps).

Main performance metrics of interest are normalized throughput and average delay. In the saturated environments, delay is the time elapsed from the moment a packet is placed at the front of the queue of an AC until the packet is successfully transmitted to the intended station, including the receipt of acknowledgement. In the unsaturated environments, delay is the time elapsed from the moment a packet arrives at the queue of an AC until the packet is successfully transmitted to the intended station, including the receipt of acknowledgement.

Figs. 6 and 7 show the results of simulation in the saturated traffic environments.

Fig. 6 shows the effect of the number of stations on normalized throughput. The proposed CFHS protocol always shows better performance than the HSAM protocol. For AC 3, as the number of stations increases, we see that performance difference between the CFHS and HSAM protocols becomes noticeable such that, in the HSAM, the throughput for AC 3 sharply decreases, whereas, in the CFHS, it slowly decreases compared with the HSAM. In the HSAM protocol, AC 3 cannot gain exclusive channel access over AC 2 and AC 1, and they can collide with one another. The HSAM has a poor normalized throughput since collision probability gets higher as the number of stations becomes larger. In the CFHS protocol, the indication phase consumes extra channel time which has a negative effect on performance. Collisions only occur while transmitting IND packets. Less channel time would be wasted when collision occurs because IND packets are very small compared with data packets. As for AC 2 and AC1, almost the same behaviours as AC 3can be observed.

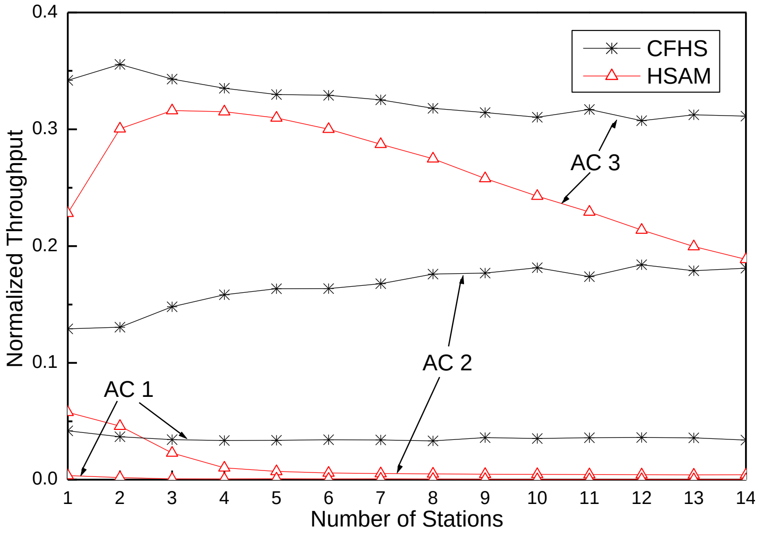

Fig. 6 .Normalized throughput according to the number stations

Fig. 7 shows the average delay based on the number of stations. Due to collisions and empty slots for the HSAM and due to the use of extra channel time in the indication phase, average delay of both protocols increases proportionally as the number of stations increases. The proposed CFHS has a lower delay compared to the HSAM.

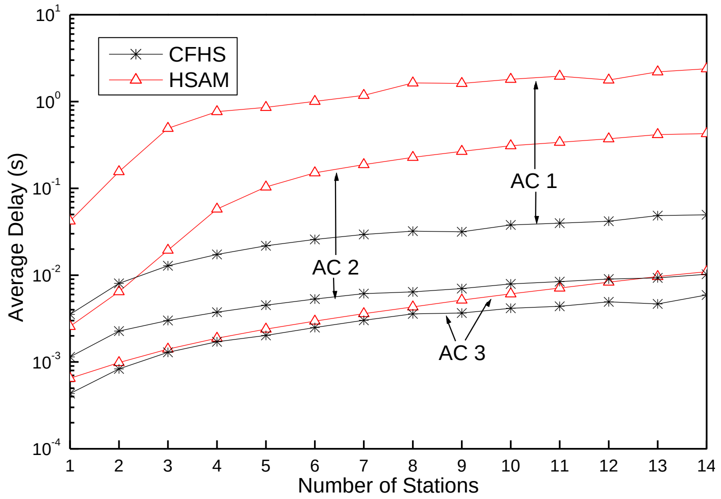

Fig. 7. Average delay according to the number stations

Figs. 8 and 9 show the results of simulation in the unsaturated traffic environments.

Fig. 8 shows the effect of the number of stations on normalized throughput. At high loads, the throughput of the HSAM becomes rapidly worse. However, the throughput of the CFHS becomes saturated at the point with a larger number of stations. From this figure, we can also observe that AC 3 obtains a significant portion of the channel time.

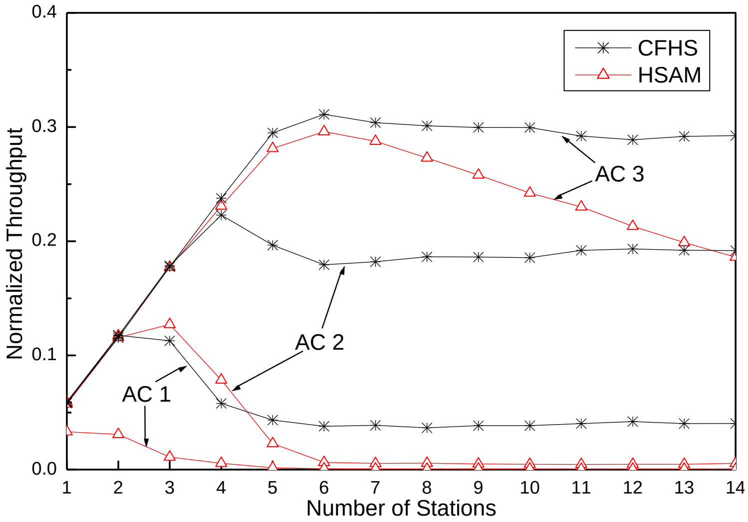

Fig. 8. Normalized throughput according to the number stations

Fig. 9 shows the average delay based on the number of stations. At light loads, the delay sharply increases as the number of stations gets larger. However, at high loads, it remains steady. The proposed CFHS always has a lower delay compared to the HSAM.

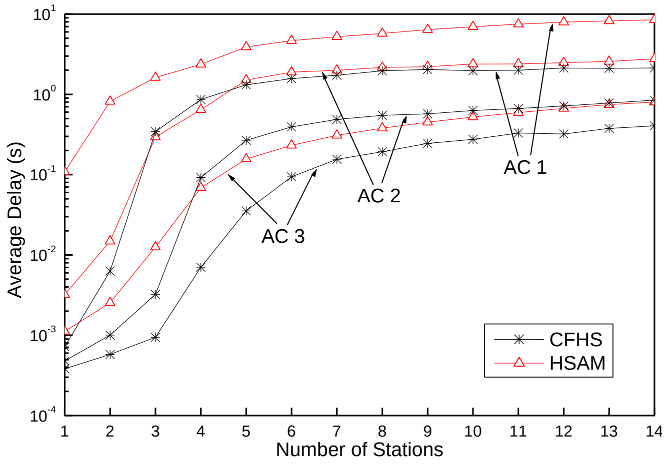

Fig. 9. Average delay according to the number stations

V. CONCLUSION

IEEE 802.11e, IEEE 802.11aa, and IEEE 802.11p use similar EDCA mechanism to provide QoS. The EDCA mechanism can be inefficient for providing QoS such as high throughput and low channel access delay because of high collision probability in networks with many stations. In this paper, we proposed a CFHS protocol to resolve collisions. The proposed CFHS protocol allocates collision free time slots for transmitting data packets to ACs which sent IND packets successfully. Therefore, it can avoid collisions and improve network performance.

Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2015R1D1A1A01059644).

REFERENCES

[1] IEEE, Wireless LAN medium access control (MAC) and physical layer (PHY) specifications, IEEE Standard 802.11-2012 (Revision of IEEE Standard 802.11-2007), 2012.

[2] Y. Xiao, A simple and effective priority scheme for IEEE 802.11, IEEE Communications Letters, 7(2), 2003, 70-72.

[3] H. Zhai, X. Chen, Y. Fang, How well can the IEEE 802.11 wireless LAN

support quality of service?, IEEE Transactions on Wireless Communications, 4(6), 2005, 3084-3094.

[4] IEEE, Part 11: wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment: medium access control (MAC) quality of service enhancements, IEEE Standard 802.11e, 2005.

[5] IEEE, Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 2: MAC enhancements for robust audio video streaming, IEEE Standard 802.11aa-2012 (Amendment to IEEE Standard 802.11-2012), 2012.

[6] K. Kosek-Szott, A. Krasilov, A. Lyakhov, M. Natkaniec, A. Safonov, S. Szott, and I. Tinnirello, What's new for QoS in IEEE 802.11?, IEEE Network, 27(6), 2013, 95-104.

[7] Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications - amendment 6: wireless access in vehicular environments. IEEE Std 802.11p-2010, 2010.

[8] M. Khatua and S. Misra, D2D: delay-aware distributed dynamic adaptation of contention window in wireless networks, IEEE Transactions on Mobile Computing, 15(2), 2016, 322-335.

[9] Y. Gao, X. Sun, and L. Dai, IEEE 802.11e EDCA networks: modeling, differentiation and optimization, IEEE Transactions on Wireless Communications, 13(7), 2014, 3863-3879.

[10] C.-Y. Chang, H.-C. Yen, and D.-J. Deng, V2V QoS guaranteed channel access in IEEE 802.11p VANETs, IEEE Transactions on Dependable and Secure Computing, 13(1), 2016, 5-17.

[11] J. Zheng and Q. Wu, Performance modeling and analysis of the IEEE 802.11p EDCA mechanism for VANET, IEEE Transactions on Vehicular Technology, 65(4), 2016, 2673-2687.

[12] A. Alonso1 and C. F. Mecklenbrauker, Dependability of decentralized congestion control for varying VANET density, IEEE Transactions on Vehicular Technology, 65(11), 2016, 9153-9167.

[13] K. Kosek-Szott, M. Natkaniec, and L. Prasnal, A novel IEEE 802.11aa intra-AC prioritization method for video transmissions, IEEE Globecom, 2014, 1158-1163.

[14] Y. Zeng, Performance analysis of the 802.11aa intra-access category prioritization under saturated condition, IEEE Globecom, 2014, 4610-4615.

[15] A. Banchs, A. Oliva, L. Eznarriaga, D. R. Kowalski, and P. Serrano, Performance analysis and algorithm selection for reliable multicast in IEEE 802.11aa wireless LAN, IEEE Transactions on Vehicular Technology, 63(8), 2014, 3875-3891.

[16] X. Yao, W. Wang, T. Wu, X. Yao, and S. Yang, A novel hybrid slot allocation mechanism for 802.11e EDCA protocol, Information Processing Letters, 114(1-2). 2014, 25-30.

[17] Y. Gao, L. Oaf, and X. Hei, Throughput optimization of non-real-time flows with delay guarantee of real-time flows in WLANs, IEEE ICC, 2015, 1541-1546.

[18] Z. J. Haas and J. Deng, Dual busy tone multiple access (DBTMA)-a multiple access control scheme for ad hoc networks, IEEE Transactions on Communications, 50(6), 2002, 975-985.

[19] K. Kang, H. Jiang, and X. Lin, Research on multi-slot virtual collision mechanism for IEEE 802.11 DCF, IEEE 68th Vehicular Technology Conference (VTC 2008-Fall), 2008, 1-6.