

RESEARCH ARTICLE

A-MPDU aggregation with optimal number of MPDUs for delay requirements in IEEE 802.11ac

Won Hyung Lee¹, Ho Young Hwang^{2*}

1 Department of Computer Engineering, Kwangwoon University, Seoul, Korea, **2** School of Computer and Information Engineering, Kwangwoon University, Seoul, Korea

* hyhwang@kw.ac.kr



OPEN ACCESS

Citation: Lee WH, Hwang HY (2019) A-MPDU aggregation with optimal number of MPDUs for delay requirements in IEEE 802.11ac. PLoS ONE 14(3): e0213888. <https://doi.org/10.1371/journal.pone.0213888>

Editor: Yanrong Lu, Tianjin University, CHINA

Received: July 6, 2017

Accepted: March 5, 2019

Published: March 20, 2019

Copyright: © 2019 Lee, Hwang. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: All relevant data are within the paper and the public repository GitHub at the following URL: <https://github.com/kw-ict/A-MPDU-802.11ac-delay>.

Funding: This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2018R1D1A1B07049601) and by the Research Grant of Kwangwoon University in 2018. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Abstract

In this paper, we propose a method that estimates an average delay of frames for each queue and finds an optimal number of aggregated Medium Access Control (MAC) Protocol Data Units (MPDUs) to maximize the system throughput with satisfying the delay requirement of each queue when using the Aggregate MPDU (A-MPDU) aggregation in IEEE 802.11ac. The delay is defined as the sum of the queuing delay and the service delay. If few frames in a queue are aggregated, the frames which remain in the queue for next transmissions may violate the target delay because of the overhead for the next transmissions such as the backoff time, Physical Layer Convergence Procedure (PLCP) preamble, and PLCP header. If many of the frames in the queue are aggregated, the frames of the queue and the other queues may violate their target delays because of a long transmission duration and a long channel occupancy. In this paper, we obtain the average delay for each queue and the optimal number of aggregated MPDUs for the delay requirement of each queue in IEEE 802.11ac. At the last, we evaluate and show the performance of our proposed method through simulations. The simulation results show that the proposed method can estimate the average delay for each queue accurately. The simulation results also show that the proposed method can obtain the violation rates on the target delays less than 0.1. Furthermore, the simulation results show that the proposed method can yield higher system throughput than other conventional methods.

Introduction

IEEE 802.11 Distributed Coordination Function (DCF) which is based on Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) in Wireless Local Area Networks (WLANs) has been developed [1, 2]. When stations (STAs) listen to the transmissions that occur from any other STAs, the STAs pause their backoff counters until the transmissions are not sensed for the Distributed Inter-Frame Space (DIFS) duration. The IEEE 802.11 Medium Access Control (MAC) manages the retransmission of collided frames with the Binary Exponential Backoff (BEB) mechanism. Request-to-Send/Clear-to-Send (RTS/CTS) method in IEEE 802.11

Competing interests: The authors have declared that no competing interests exist.

DCF is used for reducing the penalty for the collision. To evaluate the throughput performance in IEEE 802.11 DCF, Bianchi [3] analyzed the system throughput in various aspects under a saturated condition with the basic and RTS/CTS methods in IEEE 802.11 DCF with a Markov chain model. Wu et al. [4] analyzed the system performance under the saturated condition with the basic and RTS/CTS methods in IEEE 802.11 DCF with considering the frame retry limit.

Since it is difficult to service the voice traffic and video traffic differentially in IEEE 802.11 DCF, IEEE 802.11e amendment has been developed. IEEE 802.11e Enhanced Distributed Channel Access (EDCA) exploits four Access Categories (ACs) according to traffics and priorities. Each AC has maximum and minimum Contention Windows (CWs), the Arbitration Inter-Frame Space (AIFS) which is used instead of the DIFS, and Transmission Opportunity (TXOP). It makes different traffics to be serviced with supporting Quality of Service (QoS). A Block Acknowledgment (BA) mechanism is introduced in IEEE 802.11e [1, 2]. In [5], the performance of IEEE 802.11e EDCA for different CWs and AIFSs according to four ACs is analyzed. Sthapit and Pyun [6] studied an implicit BA mechanism which does not transmit the BA request frame for improving the efficiency of BA mechanism in IEEE 802.11e.

IEEE 802.11n increases the bit rate up to 150 Mbps with a single spatial stream. The Multiple-Input Multiple-Output (MIMO) technology increases the bit rate up to 600 Mbps with four spatial streams. The frame aggregation in the MAC layer can reduce the overhead such as the backoff time, Physical Layer Convergence Procedure (PLCP) preamble, and PLCP header. There are three kinds of the frame aggregation as the Aggregate MAC Service Data Unit (A-MSDU), the Aggregate MAC Protocol Data Unit (A-MPDU), and the Two-level aggregation which uses both mentioned aggregations together [1, 2]. Skordoulis et al. [7] studied the MAC enhancements and analyzed the performance of the three frame aggregation schemes in IEEE 802.11n. Liu et al. [8] studied the Two-level aggregation to improve the throughput with considering overheads for the aggregation and retransmission in IEEE 802.11n.

IEEE 802.11ac is an amendment of the IEEE 802.11 standard for improving the performance of IEEE 802.11 in WLANs. It provides some features for supporting Very High Throughput (VHT). It mandatorily supports 80 MHz channel bandwidth and optionally supports 160 MHz channel bandwidth. For 160 MHz channel bandwidth, one 160 MHz channel or two discontinuous 80 MHz channels can be utilized. It also supports the 256-QAM modulation which is the Modulation and Coding Scheme (MCS) 8 or 9. The MIMO technology and the frame aggregation in IEEE 802.11n are also supported in IEEE 802.11ac. However, IEEE 802.11ac supports up to eight spatial streams while IEEE 802.11n supports up to four spatial streams. The maximum length of MPDU and the maximum length of PLCP Service Data Unit (PSDU) in IEEE 802.11ac are 11454 bytes and 1048575 bytes, respectively, while those in IEEE 802.11n are 7935 bytes and 65535 bytes [1, 2, 9].

In A-MPDU aggregation, an A-MPDU consists of A-MPDU subframes and it can include up to 64 A-MPDU subframes. The frame format for A-MPDU is shown in Fig 1. To form an A-MPDU subframe, an MPDU delimiter is prepended to an MPDU and pad bits are appended to the MPDU. The pad bits are appended for making the length of A-MPDU subframe a multiple of 4 bytes. The pad bits are not appended to the last A-MPDU subframe in IEEE 802.11n while those are appended in IEEE 802.11ac. To form an MPDU, a MAC header is prepended to an MSDU and a Frame Check Sequence (FCS) is appended to the MSDU. If the VHT PLCP Protocol Data Unit (PPDU) duration is longer than the maximum VHT PPDU duration, 5.484 msec, a VHT STA cannot transmit the VHT PPDU [1, 9–11].

Our major contributions are as follows. We propose a method that estimates an average delay of MPDUs for each queue with A-MPDU aggregation and finds an optimal number of aggregated MPDUs for each queue to maximize the system throughput with satisfying the

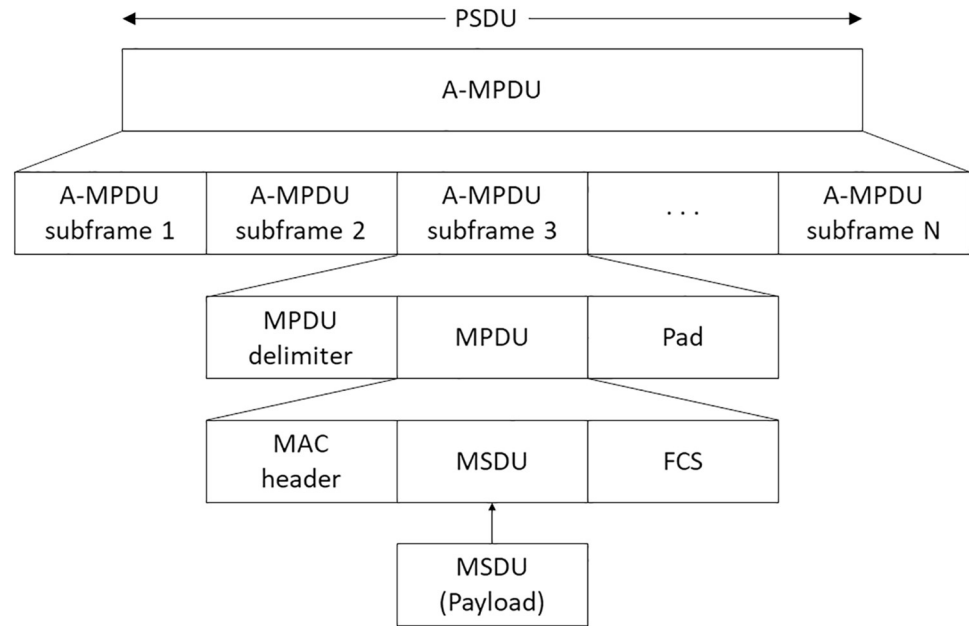


Fig 1. The frame format for A-MPDU [1, 9–11].

<https://doi.org/10.1371/journal.pone.0213888.g001>

delay requirement of each queue in IEEE 802.11ac. In this paper, we define the term queuing delay of a frame as the duration from when the frame arrived at a queue to when the queue transmits the first RTS for the frame. We define the term service delay of a frame as the duration from when a queue transmits the first RTS for the frame to when the queue receives BA for the frame. We define the term delay for each queue as the sum of the queuing delay and the service delay for each queue [12–14]. Our proposed method has two steps. In the first step, the proposed method estimates the average delay of MPDUs for each queue with considering some parameters such as the number of queues, average interval between frame arrivals at each queue, the length of each queue, minimum and maximum CWs for each queue, and the target number of aggregated MPDUs for each queue in IEEE 802.11ac. To consider the finite length of each queue, we assume that newly arrived frames at the queue are dropped when the queue is full [15]. We consider that the frame arrival rate of each queue is fast enough. The proposed method can pop MPDUs from the queue to aggregate the MPDUs when transmitting the first RTS for the MPDUs. From that time, the MPDUs can wait at the extra memory until the reception of BA for the successful transmission of A-MPDU [16–18]. To estimate the average delay of MPDUs for each queue, we analyze the average interval between two consecutive successful transmissions from each queue and the average duration from when receiving BA until transmitting the first RTS for the next aggregated MPDUs from each queue in IEEE 802.11ac. In the second step, the proposed method finds the optimal number of aggregated MPDUs for each queue to maximize the system throughput with satisfying the delay requirement of each queue using the estimated average delay for each queue. If there are multiple optimal cases with ties, the proposed method selects the case with the lowest standard deviation of ratios of average delays to target delays for the queues. Through simulations, we verify that the proposed method is valid. Simulation results show that the proposed method can estimate the average delay for each queue accurately. The analytical results of the proposed method are close to simulation results when the frame arrival process follows a Poisson distribution. The simulation results also show that the proposed method can obtain the violation

rates on the target delays less than 0.1. Furthermore, the simulation results show that the proposed method can yield higher system throughput than other conventional methods.

Related works

In this section, we introduce several studies for IEEE 802.11 WLANs. Since IEEE 802.11 is the most promising mechanism in WLANs, there are many papers to improve the throughput and delay performance in IEEE 802.11 WLANs. Kim and Cho [19] studied an Adaptive TXOP Allocation (ATA) scheme for satisfying the delay bound of each packet in a queue in IEEE 802.11e EDCA. The ATA scheme checks whether each packet in a queue satisfies its delay bound or not with increasing TXOP before every transmission. The ATA scheme needs additional information about network load from an Access Point (AP) through the beacon frame. They showed the performance of the scheme through a simulation wherein an STA with multimedia traffic uses the ATA scheme while other STAs do not use TXOP. In [20], a method which finds optimal TXOP for throughput requirements under a saturated condition was studied in IEEE 802.11e EDCA. The method divided queues into two sets, one for feasible queues, another for unfeasible queues in terms of throughput requirements. The queues which satisfy their throughput requirements decrease their TXOP, and the queues which do not satisfy their throughput requirements increase their TXOP. Simulation results show that the method makes throughputs of queues close to their throughput requirements. In [19] and [20], the schemes using TXOP were studied to satisfy the throughput requirements or the delay requirements in IEEE 802.11e EDCA. In this paper, our proposed method utilizes A-MPDU aggregation instead of TXOP to satisfy the delay requirement of each queue in IEEE 802.11ac.

In [10], the performance of the A-MSDU aggregation and A-MPDU aggregation for different aggregation sizes and frame arrival rates was analyzed in IEEE 802.11n. The arrival process of frames was considered as a Poisson distribution. A Discrete Time Markov Chain (DTMC) model under an unsaturated condition was studied to have the states which describe the frame aggregation. Karmakar et al. [21] analyzed the throughput of A-MPDU aggregation in IEEE 802.11n under an unsaturated condition. They take account of A-MPDU aggregation level and data generation probability. They also analyzed MAC access delay. In this paper, our proposed method estimates the average delay for each queue which includes the MAC access delay for the queue. To estimate the average delay of MPDUs for each queue, we analyze the average interval between two consecutive successful transmissions from each queue and the average duration from when receiving BA until transmitting the first RTS for the next aggregated MPDUs from each queue in IEEE 802.11ac.

Jin et al. [22] analyzed the throughput performance with the Single-User MIMO (SU-MIMO) and the Multi-User MIMO (MU-MIMO) in uplink WLANs. Their simulation results show that if the number of STAs is small, the throughput of the SU-MIMO is higher than that of the MU-MIMO. Sharon and Alpert [9] compared the maximum throughputs of the A-MSDU aggregation, A-MPDU aggregation, and Two-level aggregation in IEEE 802.11n and IEEE 802.11ac for an error-free channel and an error-prone channel. They studied that the maximum throughput of IEEE 802.11ac is larger than or equal to that of IEEE 802.11n because of the maximum lengths of MPDU, A-MSDU, and A-MPDU. In [11], they also analyzed the performance of A-MPDU aggregation and Two-level aggregation with and without the RTS/CTS method when homogenous ACs are active in IEEE 802.11ac. They considered the transmission delay limit instead of the retransmission count limit. In their simulations, the ACs can retransmit failed frames whenever the frames are within their delay limit. They showed the maximum AppRate which is the summation of data rates for each AC with varying

the number of MPDUs in PSDU and the number of ACs. In this paper, we consider the delay requirement of each queue with A-MPDU aggregation in IEEE 802.11ac. To satisfy the delay requirement of each queue, we estimate the average delay of MPDUs for each queue with considering some parameters such as the number of queues, average interval between frame arrivals at each queue, the length of each queue, minimum and maximum CWs for each queue, and the target number of aggregated MPDUs for each queue in IEEE 802.11ac.

A-MPDU aggregation with optimal number of MPDUs for delay requirements in IEEE 802.11ac

In this section, we propose a method that estimates the average delay of MPDUs for each queue and finds the optimal number of aggregated MPDUs to maximize the system throughput with satisfying the delay requirement of each queue when the RTS/CTS method is used in IEEE 802.11ac. Our proposed method has two steps. In the first step, the proposed method estimates the average delay of MPDUs for each queue with considering some parameters such as the number of queues, average interval between frame arrivals at each queue, the length of each queue, minimum and maximum CWs for each queue, and the target number of aggregated MPDUs for each queue in IEEE 802.11ac. To consider the finite length of each queue, we assume that newly arrived frames at the queue are dropped when the queue is full [15]. We consider that the frame arrival rate of each queue is fast enough. The proposed method can pop MPDUs from the queue to aggregate the MPDUs when transmitting the first RTS for the MPDUs. From that time, the MPDUs can wait at the extra memory until the reception of BA for the successful transmission of A-MPDU [16–18]. To estimate the average delay of MPDUs for each queue, we analyze the average interval between two consecutive successful transmissions from each queue and the average duration from when receiving BA until transmitting the first RTS for the next aggregated MPDUs from each queue in IEEE 802.11ac. In the second step, the proposed method finds the optimal number of aggregated MPDUs for each queue to maximize the system throughput with satisfying the delay requirement of each queue using the estimated average delay for each queue. If there are multiple optimal cases with ties, the proposed method selects the case with the lowest standard deviation of ratios of average delays to target delays for the queues.

We consider A-MPDU aggregation of a queue with a fixed number of MPDUs for each transmission from the queue. Since we focus on the performance of A-MPDU aggregation, we assume that the queues have the same AIFS and there are N queues in a system [20]. The main notations are shown in Table 1. Let $T_{w,i}$ denote the average waiting time for a successful transmission from the i -th queue, $i = 1, 2, \dots, N$. Then, $T_{w,i}$ can be expressed as:

$$T_{w,i} = T_i - (A_i + O_{tx} + AIFS), i \in [1, N], \tag{1}$$

where $AIFS = \delta \cdot AIFSN + SIFS$ and δ is the length of the timeslot which is empty [1]. $[a, b]$ is the set of integers, $\{x|a \leq x \leq b\}$ for integers a and b . The average interval between two consecutive successful transmissions from the i -th queue T_i is defined as:

$$T_i = \frac{(1 - p_b) \cdot \delta + p_s \cdot T_s + (p_b - p_s) \cdot T_c}{p_{s,i}}, i \in [1, N], \tag{2}$$

where $p_b, p_{s,i}, p_s, T_s$, and T_c are defined in [20]. $p_b, p_{s,i}$ and p_s are the probability that the status of the channel is sensed as busy, the probability that a successful transmission from the i -th queue occurs, and the probability that a successful transmission occurs in the channel, respectively. T_s and T_c are the average times that a successful transmission and a collision occur, respectively, in the channel. Let $T_{service}, T_{payload}$, and T_{tail} be the transmission durations for

Table 1. Notations.

Notation	Meaning
N	The number of queues
p_b	The probability that the status of the channel is sensed as busy
$p_{s,i}$	The probability that a successful transmission from the i -th queue occurs
p_s	The probability that a successful transmission occurs in the channel
$p_{b,Ni}$	The probability that the status of the channel is sensed as busy during the backoff procedure of the i -th queue
$p_{s,i}$	The probability that a successful transmission occurs in the channel during the backoff procedure of the i -th queue
A_i	The duration for transmission of A-MPDU from the i -th queue
F_i	The target number of aggregated MPDUs for the i -th queue
I_i	The average interval between frame arrivals at the i -th queue
D_i	The average delay for the i -th queue
D_{ij}	The average delay of the j -th frame among F_i frames for the i -th queue
$T_{w,i}$	The average waiting time for a successful transmission from the i -th queue
T_i	The average interval between two consecutive successful transmissions from the i -th queue
T_s	The average times that a successful transmission occurs in the channel
T_c	The average times that a collision occurs in the channel
$T_{service}$	The transmission duration of PLCP service field
$T_{payload}$	The transmission duration of payload
T_{tail}	The transmission duration of PLCP tail field
T_{RTS}	The transmission duration of RTS
T_{CTS}	The transmission duration of CTS
$T_{preamble}$	The transmission duration of PLCP preamble
T_{BA}	The transmission duration of BA
δ	The length of the timeslot which is empty
$T_{slot,i}$	The average duration of a virtual slot during the backoff procedure of the i -th queue.
$T_{BO,i}$	The average duration from when receiving BA until transmitting the first RTS for the next aggregated MPDUs from the i -th queue
W_i	The minimum CW of the i -th queue
L_i	The length of the i -th queue
O_{tx}	The overhead time for a successful transmission
O_{A-MPDU}	A-MPDU overhead time for MPDU delimiter, MAC header, FCS, and pad bits

<https://doi.org/10.1371/journal.pone.0213888.t001>

PLCP service field, payload, and PLCP tail field, respectively. Then, the duration for transmission of A-MPDU from the i -th queue, A_i can be expressed as:

$$A_i = T_{service} + F_i \cdot (T_{payload} + O_{A-MPDU}) + T_{tail}, i \in [1, N], \tag{3}$$

where F_i is the target number of aggregated MPDUs for the i -th queue, O_{A-MPDU} is A-MPDU overhead time for MPDU delimiter, MAC header, FCS, and pad bits. In Eq (1), the overhead time for a successful transmission O_{tx} can be expressed as:

$$O_{tx} = T_{RTS} + 3 \cdot SIFS + T_{CTS} + T_{preamble} + T_{BA}, \tag{4}$$

where T_{RTS} , T_{CTS} , $T_{preamble}$, and T_{BA} are the transmission durations for RTS, CTS, PLCP preamble, and BA, respectively.

We assume that newly arrived frames at a queue are dropped when the queue is full [15]. Let I_i denote the average interval between frame arrivals at the i -th queue. We assume that

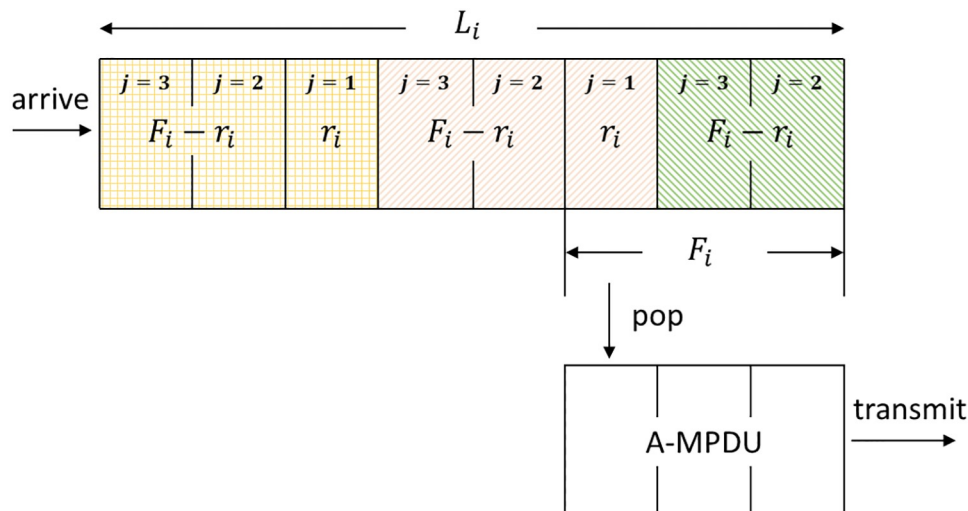


Fig 2. Example of a status for a queue [16–18].

<https://doi.org/10.1371/journal.pone.0213888.g002>

each frame arrives with the same interval I_i like Constant Bit Rate (CBR) traffic [23]. Let D_i denote the average delay for the i -th queue. Then, D_i can be expressed as:

$$D_i \simeq \sum_{j=1}^{F_i} \frac{D_{ij}}{F_i}, i \in [1, N], \tag{5}$$

where D_{ij} is the average delay of the j -th frame among F_i frames for the i -th queue. The proposed method can pop F_i MPDUs from the queue to aggregate the MPDUs when transmitting the first RTS for the MPDUs as shown in Fig 2. From that time, the MPDUs can wait at the extra memory until the reception of BA for the successful transmission of A-MPDU [16–18]. Then, D_{ij} can be obtained as:

$$D_{ij} \simeq \begin{cases} T_i \cdot (Q_i - 1) + T_{w,i} + O_{tx} + A_i - T_{BO,i} - I_i \cdot (j - \frac{1}{2}), j \in [1, r_i] \\ T_i \cdot Q_i + T_{w,i} + O_{tx} + A_i - T_{BO,i} - I_i \cdot (j - \frac{1}{2}), j \in [r_i + 1, F_i] \end{cases}, i \in [1, N], \tag{6}$$

where $T_{BO,i}$, $T_{slot,i}$, Q_i , and r_i can be expressed as:

$$T_{BO,i} = \frac{W_i - 1}{2} \cdot T_{slot,i}, i \in [1, N], \tag{7}$$

$$T_{slot,i} = (1 - p_{b,\setminus i}) \cdot \delta + p_{s,\setminus i} \cdot T_s + (p_{b,\setminus i} - p_{s,\setminus i}) \cdot T_c, i \in [1, N], \tag{8}$$

$$Q_i = \left\lfloor \frac{L_i}{F_i} \right\rfloor + 1, i \in [1, N], \tag{9}$$

$$r_i = F_i - (L_i \bmod F_i), i \in [1, N], \tag{10}$$

where $\lfloor n \rfloor$ is the largest integer smaller than or equal to real number n . $(a \bmod b)$ is the remainder of a divided by b . $p_{b,\setminus i}$ and $p_{s,\setminus i}$ are the probability that the status of the channel is sensed as busy and the probability that a successful transmission occurs in the channel, respectively, during the backoff procedure of the i -th queue [24]. $T_{slot,i}$ is the average duration of a

virtual slot [5] during the backoff procedure of the i -th queue. To obtain $T_{w,i}$, T_i , and $T_{slot,i}$ we use Wu's model [4]. Some parameters such as the minimum and maximum CWs of the i -th queue affect $T_{w,i}$, T_i , and $T_{slot,i}$. $T_{BO,i}$ is the average duration from when receiving BA until transmitting the first RTS for the next aggregated MPDUs from the i -th queue. W_i is the minimum CW of the i -th queue. L_i is the length of the i -th queue, i.e., the maximum number of frames which can be queued at the i -th queue.

Fig 2 shows an example of a status for a queue where the length of the queue is 8 and the target number of aggregated MPDUs is 3. That is, $L_i = 8$ and $F_i = 3$, so $r_i = 3 - (8 \bmod 3) = 1$. In Fig 2, for the three frames which are aggregated and transmitted, let's assume that the first two frames have arrived after the g -th successful transmission, then the last one frame has arrived after the $(g + 1)$ -th successful transmission. Thus, we can estimate the average delay for the i -th queue by dividing F_i frames into two sets which consist of $F_i - r_i$ frames and r_i frames, respectively. In this paper, we consider that the frame arrival rate $1/I_i$ is fast enough. Otherwise, the actual average delay for the i -th queue may be less than D_i since Eqs (1), (2), and (7) to (10) are obtained under a saturated condition.

Now, we find the optimal number of aggregated MPDUs for each queue to maximize the system throughput with satisfying the delay requirement of each queue in IEEE 802.11ac. The proposed method estimates the average delay of each queue to satisfy the delay requirement of each queue. The average delay of each MPDU D_{ij} depends on the target number of aggregated MPDUs F_i and the length of each queue L_i . Thus, we propose a method to find an optimal number of aggregated MPDUs for each queue F_i^* to satisfy the delay requirement of each queue when the frame arrival interval I_i and the length of each queue L_i are given. It is possible to simply guess that the average delay of each MPDU for the i -th queue should be less than the target delay of the i -th queue d_i . Then, we can obtain the constraint (11) with the number of aggregated MPDUs for each queue $\mathbf{F} = (F_1, F_2, \dots, F_N)$ as:

$$D_{ij}(\mathbf{F}) \leq d_i, j \in [1, F_i], i \in [1, N]. \tag{11}$$

However, if D_{ij} is close to d_i , a half of the transmitted MPDUs may violate the target delay of the i -th queue since D_{ij} is an average value. Thus, we use the constraint (12) instead of the constraint (11) with considering the weight value ω_i which is the real value ($0 < \omega_i \leq 1$). The constraint (12) means that the average delay of the i -th queue should be less than or equal to the weighted target delay of the i -th queue as:

$$D_i(\mathbf{F}) \leq \omega_i \cdot d_i, i \in [1, N]. \tag{12}$$

Moreover, to maximize the system throughput with satisfying the constraint (12), we use Eq (13) as:

$$F_{th} = \operatorname{argmax}_F \sum_{i=1}^N F_i. \tag{13}$$

The large number of aggregated MPDUs for each queue can cause a high system throughput since we consider the RTS/CTS method. If there is more than one value F_{th} with ties from Eqs (13) and (14) is used to select one of them as:

$$F^* = \operatorname{argmin}_{F_{th}} \operatorname{std}(\mathbf{D}_{norm}), \tag{14}$$

where $\mathbf{D}_{norm} = (D_1/d_1, D_2/d_2, \dots, D_N/d_N)$ and $\operatorname{std}(\mathbf{x})$ is the standard deviation of elements of vector \mathbf{x} . Through Eqs (13) and (14), we can obtain $\mathbf{F}^* = (F_1^*, F_2^*, \dots, F_N^*)$ by considering not

only the summation of elements of vector F but also the standard deviation of ratios of the average delays to the target delays for the queues D_{norm} .

Simulation results

In this section, we evaluate the performance of our proposed method through simulations. First, we evaluate the performance of our proposed method in terms of the average delay with A-MPDU aggregation. Then, we evaluate the performance of the proposed method in terms of the violation rate on the target delay. We use Network Simulator 3 (NS-3) version 26 for simulations [18]. The input parameters for simulations are shown in Table 2 [1, 18]. In NS-3, when the size of A-MPDU is smaller than the maximum A-MPDU size, the MPDUs for a queue can be aggregated up to 64 MPDUs which is given as the maximum number of aggregated MPDUs in NS-3 simulations. We set the maximum A-MPDU size in NS-3 enough to aggregate the MPDUs as the target number of aggregated MPDUs. In NS-3, each queue may aggregate the MPDUs less than the target number of aggregated MPDUs whenever it has less MPDUs than the target number of aggregated MPDUs. This may cause the difference between the analytical results and the simulation results. We consider that each STA has a queue with best effort traffic. We run NS-3 simulations 10 times for each point of each figure in this section and we show the average and the standard deviation of the simulation results in each figure.

Fig 3 shows the average delay with varying the target number of aggregated MPDUs. To validate the proposed model, we run the simulation in NS-3 with input parameter set I when the numbers of STAs are 5 and 10, respectively. To compare the analytical results with the simulation results, we vary the target number of aggregated MPDUs as 4, 8, 16, 32, and 64. In Fig 3, the analytical results are shown as lines and the averages of the simulation results are shown as symbols. The error bar on each symbol indicates the standard deviation of the simulation results. When the number of STAs is 5, the analytical results are 117.09, 89.85, 77.64, 73.99, and 77.00 msec, while the simulation results are 109.5, 85.53, 74.63, 70.88, and 72.42 msec. When the number of STAs is 10, the analytical results are 240.39, 183.98, 159.52, 154.03 and 164.64 msec, while the simulation results are 217.27, 171.10, 150.93, 146.01, and 153.54 msec. Through the NS-3 simulations, the results show that the proposed method can estimate the average delay accurately.

Let $N_i^{violated}$, N_i^{total} , R_c , and R_{avg} denote the number of MPDUs which violate the target delay of the i -th STA, the number of transmitted MPDUs from the i -th STA, the violation rate on the target delay of each class $c \in [1, C]$, and the average violation rate on the target delays, respectively. Then, R_c and R_{avg} can be obtained as:

$$R_c = \frac{\sum_{i \in class[c]} N_i^{violated}}{\sum_{i \in class[c]} N_i^{total}}, c \in [1, C], \tag{15}$$

$$R_{avg} = \frac{\sum_{i=1}^N N_i^{violated}}{\sum_{i=1}^N N_i^{total}}. \tag{16}$$

For Figs 4 and 5, we run NS-3 simulations to show the violation rates on the target delays with input parameter set II. The error bar indicates the standard deviation of the simulation results. We consider three classes ($C = 3$) and each class consists of three STAs ($N = 9$). These classes have different delay requirements and frame arrival intervals. The target delay of each class $c \in [1, 3]$ is given by the vector (160, 320, 480) msec, and the frame arrival interval of each class $c \in [1, 3]$ is given by the vector (100, 200, 300) μ sec. From Eqs (1) to (14), we can obtain

Table 2. Input parameters for simulations [1, 18].

Input parameter set I	
Parameter	Value
MPDU size	502 bytes
Data rate	54 Mbps
Control rate	13.5 Mbps
The length of each queue	100
Input parameter set II	
Parameter	Value
MPDU size	1498 bytes
Data rate	180 Mbps
Control rate	13.5 Mbps
The number of STAs N	9
The number of class C	3
STA indices for class c , $c \in [1, C]$	$\{3c - 2, 3c - 1, 3c\}$
The length of each queue	150
Input parameter set III	
Parameter	Value
MPDU size	1498 bytes
Data rate	54 Mbps
Control rate	13.5 Mbps
The number of STAs N	9
The number of class C	3
STA indices for class c , $c \in [1, C]$	$\{3c - 2, 3c - 1, 3c\}$
The length of each queue	100
Common parameters	
Parameter	Value
RTS size	20 bytes
CTS size	14 bytes
BA size	32 bytes
MPDU delimiter size	4 bytes
MAC header size	26 bytes
FCS size	32 bits
PSDU service field size	16 bits
PSDU tail field size	6 bits
The minimum CW	15
The maximum CW	1023
AIFSN	3
Slot time	9 μ sec
SIFS time	16 μ sec
PLCP preamble time	40 μ sec
RTS transmission duration	36 μ sec
CTS transmission duration	44 μ sec
BA transmission duration	32 μ sec
CTS timeout	75 μ sec

<https://doi.org/10.1371/journal.pone.0213888.t002>

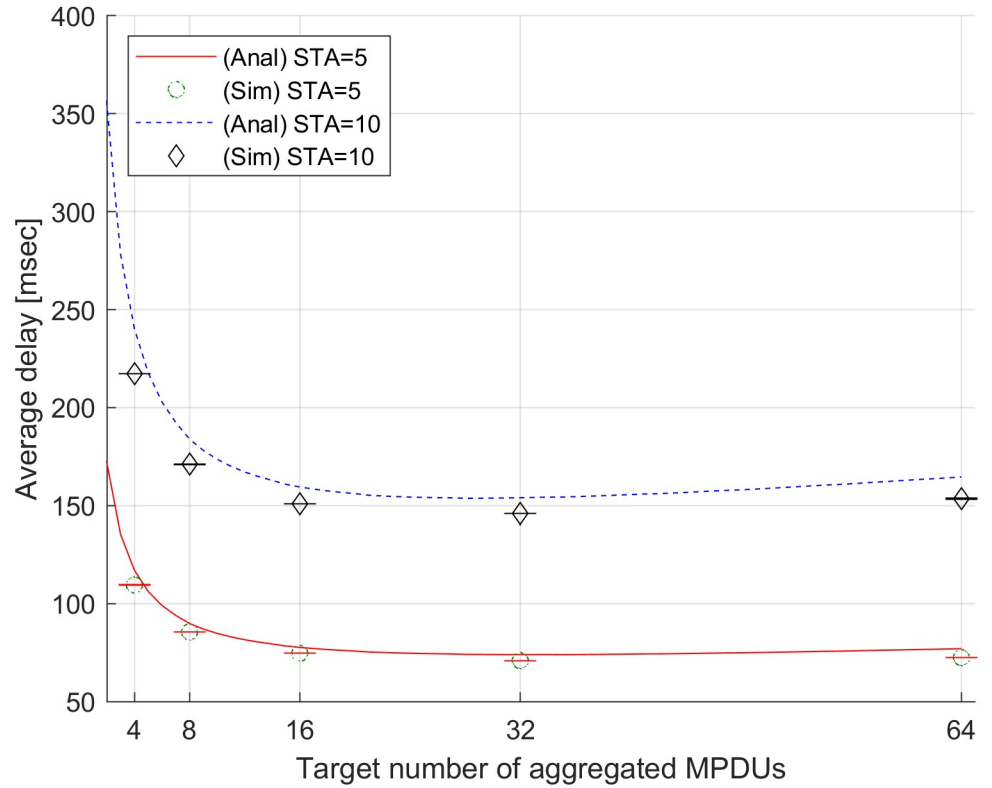


Fig 3. Average delay with varying the target number of aggregated MPDUs.

<https://doi.org/10.1371/journal.pone.0213888.g003>

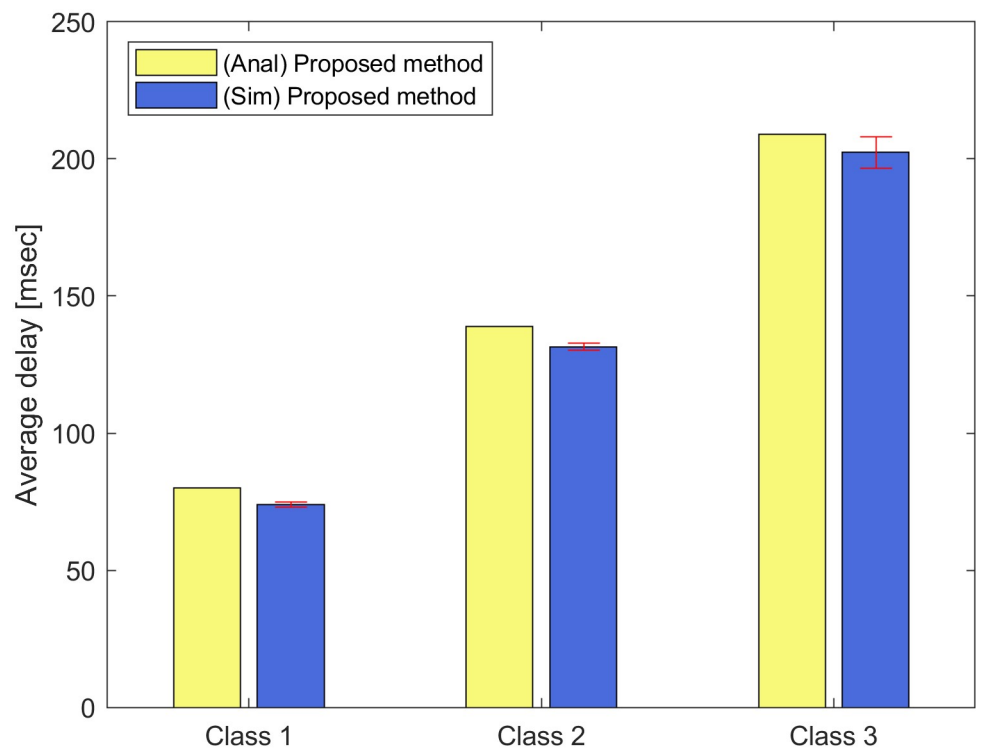


Fig 4. Average delay of each class.

<https://doi.org/10.1371/journal.pone.0213888.g004>

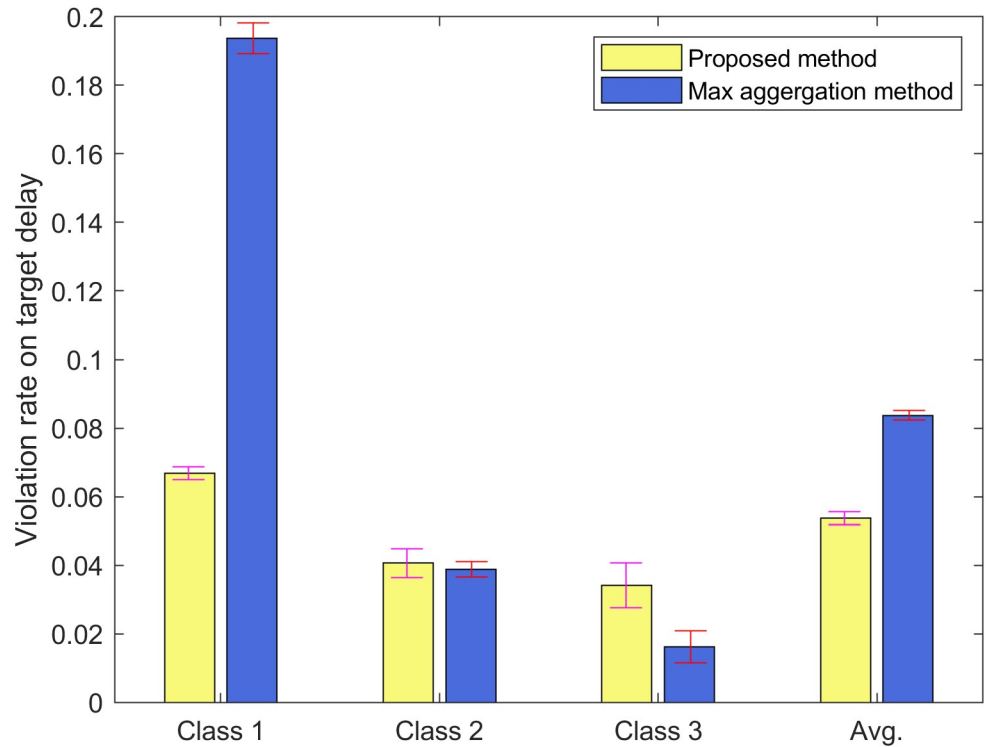


Fig 5. The violation rate on the target delay of each class.

<https://doi.org/10.1371/journal.pone.0213888.g005>

the target numbers of aggregated MPDUs for each class $c \in [1, 3]$ as 64, 33, and 21, respectively, with the weight value $\omega_i = 0.5$, $i \in [1, 9]$ in the constraint (12). Fig 4 shows the average delay of each class $c \in [1, 3]$. In Fig 4, the analytical results for the average delays of classes $c \in [1, 3]$ are 79.93, 139.00, and 208.97 msec, while the simulation results for those are 73.98, 131.47, and 202.35 msec. The simulation results are little smaller than the analytical results, but they are close.

Fig 5 shows the violation rate on the target delay of each class $c \in [1, 3]$ and the average violation rate on the target delays. In Fig 5, we also simulate the max aggregation method which aggregates the MPDUs for each queue up to 64 MPDUs to compare the performance of the proposed method with that of the max aggregation method. In Fig 5, the violation rates on the target delays of classes $c \in [1, 3]$ for the proposed method are 0.067, 0.041, and 0.034, while those for the max aggregation method are 0.194, 0.039, and 0.016. For class 1, the violation rate on the target delay for the max aggregation method is approximately three times higher than that for the proposed method. Since the target numbers of aggregated MPDUs for classes 2 and 3 of the proposed method are less than those of the max aggregation method, the proposed method can aggregate less MPDUs for classes 2 and 3 than the max aggregation method. It means that the average duration of a virtual slot during the backoff procedure for class 1 of the proposed method can be shorter than that of the max aggregation method. Thus, the average delay for class 1 of the proposed method as 73.98 msec is shorter than that of the max aggregation method as 116.34 msec with input parameter set II. Due to this reason, the violation rate on the target delay for class 1 of the proposed method is lower than that of the max aggregation method in Fig 5. On the other hand, class 3 of the proposed method may need more successful transmissions than class 3 of the max aggregation method in order to transmit the same number of MPDUs successfully due to less target number of aggregated MPDUs. Thus, the average

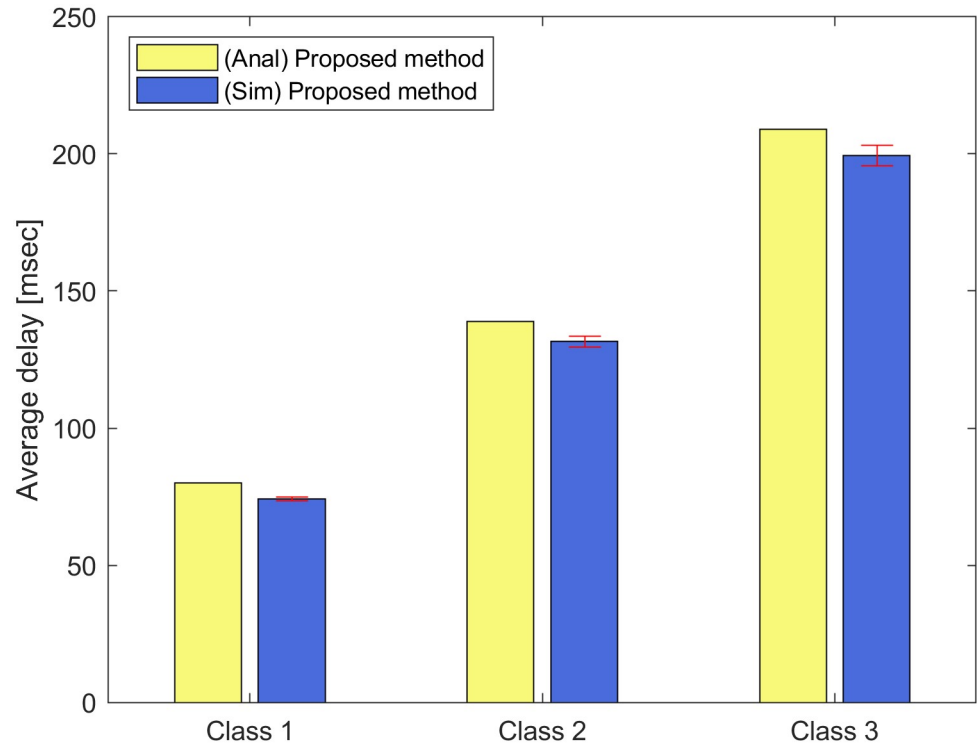


Fig 6. Average delay of each class under a Poisson distribution.

<https://doi.org/10.1371/journal.pone.0213888.g006>

delay for class 3 of the proposed method as 202.35 msec is longer than that of the max aggregation method as 109.22 msec with input parameter set II. Due to this reason, the violation rate on the target delay for class 3 of the proposed method is higher than that of the max aggregation method in Fig 5.

The results show that the class which has shorter target delay has a larger violation rate on the target delay. The average violation rates on the target delays for the proposed method and the max aggregation method are 0.054 and 0.084, respectively. The results show that the average violation rate on the target delays of the max aggregation method is higher than that for the proposed method.

For Figs 6 and 7, we run NS-3 simulations when the frame arrival process follows a Poisson distribution. We also use the input parameter set II. The error bar indicates the standard deviation of the simulation results. Fig 6 shows the average delay of each class $c \in [1, 3]$. In Fig 6, the analytical results for the average delays of classes $c \in [1, 3]$ are 79.93, 139.00, and 208.97 msec, while the simulation results for those are 74.22, 131.60, and 199.51 msec. Even though the frame arrival process follows a Poisson distribution, the analytical results of the proposed method are close to simulation results. Fig 7 shows the violation rate on the target delay of each class $c \in [1, 3]$ when the frame arrival process follows a Poisson distribution. In Fig 7, the violation rates on the target delays of classes $c \in [1, 3]$ for the proposed method are 0.066, 0.040, and 0.033, while those for the max aggregation method are 0.187, 0.042, and 0.016. The average violation rates on the target delays for the proposed method and the max aggregation method are 0.053 and 0.082, respectively. The results show that the proposed method can obtain the violation rates on the target delays less than 0.1 while the max aggregation method may not when the frame arrival process follows a Poisson distribution.

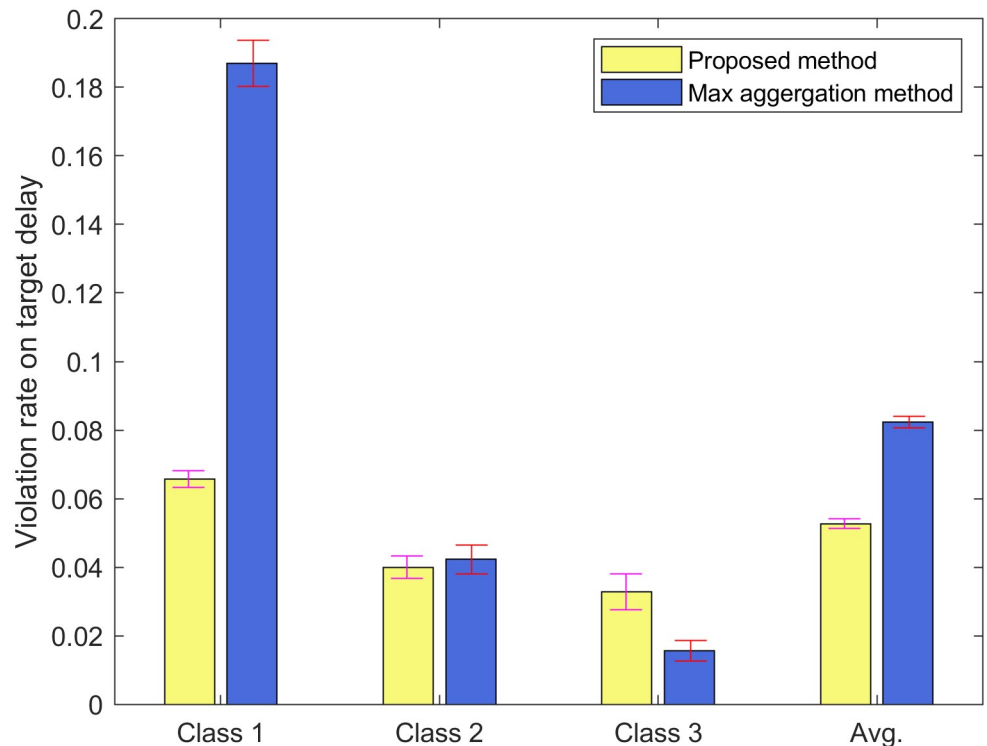


Fig 7. The violation rate on the target delay of each class under a Poisson distribution.

<https://doi.org/10.1371/journal.pone.0213888.g007>

For Figs 8 and 9, we run NS-3 simulations with input parameter set III. The error bar indicates the standard deviation of the simulation results. In Figs 8 and 9, we compare our proposed method with the max aggregation method, A-MSDU using the value of the maximum length of A-MSDU in IEEE 802.11 standard (A-MSDU with the max length in IEEE 802.11) [1], and A-MPDU using the default value of the maximum length of A-MPDU in NS-3 (A-MPDU with the default length in NS-3) [18]. The target delays for classes $c \in [1, 3]$ are same as 500 msec. From Eqs (1) to (14), we can obtain the same target number of aggregated MPDUs for classes $c \in [1, 3]$ as 20, with the weight value $\omega_i = 0.5$, $i \in [1, 9]$ in the constraint (12). For A-MSDU with the max length in IEEE 802.11, the target number of aggregated MSDUs is 7 since the maximum length of A-MSDU is 11,454 bytes according to IEEE 802.11 standard [1]. For A-MPDU with the default length in NS-3, the target number of aggregated MPDUs is 42 since the default value of the maximum length of A-MPDU for the best effort traffic in NS-3 is 65,535 bytes [18]. In Figs 8 and 9, we set the maximum VHT PPDU duration as 10 sec which is longer than that of IEEE 802.11 standard to compare the proposed method with other conventional methods. Fig 8 shows the violation rate on the target delay of each class for each method. In Fig 8, the violation rate on the target delay of the proposed method is less than 0.1. The violation rate on the target delay of the proposed method is lower than those of the max aggregation method and A-MPDU with the default length in NS-3. For A-MSDU with the max length in IEEE 802.11 with input parameter set III, the violation rate on the target delay is the lowest in Fig 8, but the system throughput is the lowest in Fig 9 since the target number of aggregated frames is the smallest among four methods. Fig 9 shows the throughput of each class and the system throughput for each method. In Fig 9, A-MSDU with the max length in IEEE 802.11 yields the lowest system throughput, while our proposed method yields the highest system throughput among four methods.

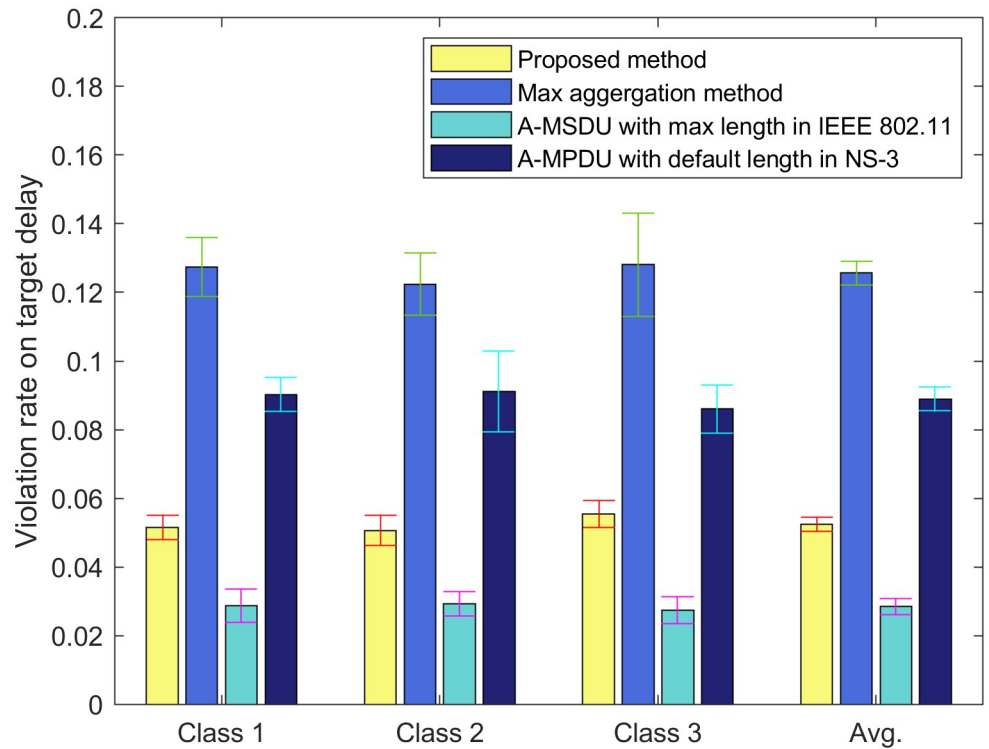


Fig 8. Comparison of the proposed method with other conventional methods for violation rate on the target delay of each class.

<https://doi.org/10.1371/journal.pone.0213888.g008>

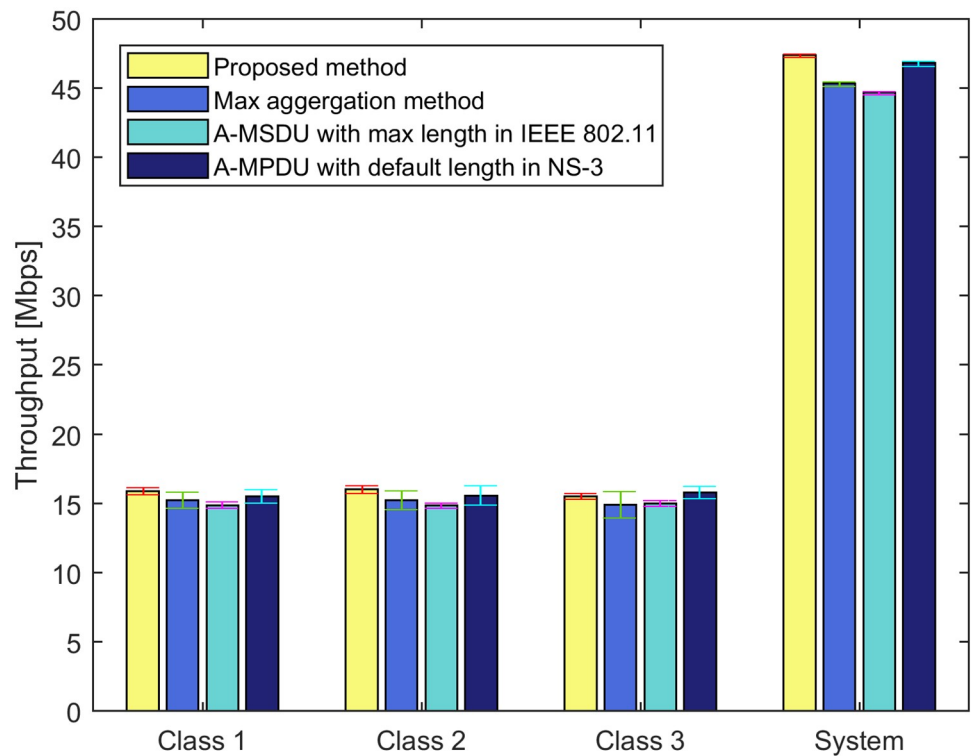


Fig 9. Comparison of the proposed method with other conventional methods for throughput of each class.

<https://doi.org/10.1371/journal.pone.0213888.g009>

Conclusion

In this paper, we proposed a method that estimates an average delay of MPDUs for each queue and finds an optimal number of aggregated MPDUs for each queue to maximize the system throughput with satisfying the delay requirement of each queue in IEEE 802.11ac. We evaluated the performance of the proposed method through NS-3 simulations. The simulation results showed that the proposed method can estimate the average delay for each queue accurately. The simulation results also showed that the proposed method can obtain the violation rates on the target delays less than 0.1 because the proposed method uses the optimal number of aggregated MPDUs which is considered average delay. We also ran NS-3 simulations when the frame arrival process follows a Poisson distribution. The analytical results of the proposed method were close to simulation results when the frame arrival process follows a Poisson distribution. We compared the proposed method with other conventional methods through NS-3 simulations. The simulation results showed that the proposed method can yield higher system throughput than other conventional methods.

Author Contributions

Conceptualization: Won Hyoung Lee, Ho Young Hwang.

Formal analysis: Won Hyoung Lee, Ho Young Hwang.

Funding acquisition: Ho Young Hwang.

Investigation: Won Hyoung Lee, Ho Young Hwang.

Methodology: Won Hyoung Lee, Ho Young Hwang.

Project administration: Ho Young Hwang.

Software: Won Hyoung Lee, Ho Young Hwang.

Supervision: Ho Young Hwang.

Validation: Won Hyoung Lee.

Visualization: Won Hyoung Lee.

Writing – original draft: Won Hyoung Lee, Ho Young Hwang.

Writing – review & editing: Won Hyoung Lee, Ho Young Hwang.

References

1. IEEE Standard 802.11, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. 2016.
2. Charfi E, Chaari L, Kamoun L. PHY/MAC Enhancements and QoS Mechanisms for Very High Throughput WLANs: A Survey. *IEEE Communications Surveys & Tutorials*. 2013; 15(4):1714–1735. <https://doi.org/10.1109/SURV.2013.013013.00084>
3. Bianchi G. Performance Analysis of the IEEE 802.11 Distributed Coordination Function. *IEEE Journal on Selected Areas in Communications*. 2000; 18(3):535–547. <https://doi.org/10.1109/49.840210>
4. Wu H, Peng Y, Long K, Cheng S, Ma J. Performance of Reliable Transport Protocol over IEEE 802.11 Wireless LAN: Analysis and Enhancement. *IEEE Information Communications (INFOCOM)*; 2002.
5. Hwang HY, Kim SJ, Sung DK, Song NO. Performance Analysis of IEEE 802.11e EDCA with a Virtual Collision Handler. *IEEE Transactions on Vehicular Technology*. 2008; 57(2):1293–1297. <https://doi.org/10.1109/TVT.2007.911611>
6. Sthapit P, Pyun JY. Implicit Block ACK Scheme for IEEE 802.11 WLANs. *Sensors*. 2016; 16(2):167. <https://doi.org/10.3390/s16020167>

7. Skordoulis D, Ni Q, Chen HH, Stephens AP, Liu C, Jamalipour A. IEEE 802.11n MAC Frame Aggregation Mechanisms for Next-Generation High-Throughput WLANs. *IEEE Wireless Communications*. 2008; 15(1):40–47. <https://doi.org/10.1109/MWC.2008.4454703>
8. Liu J, Yao M, Qiu Z. Enhanced Two-Level Frame Aggregation with Optimized Aggregation Level for IEEE 802.11n WLANs. *IEEE Communications Letters*. 2015; 19(12):2254–2257. <https://doi.org/10.1109/LCOMM.2015.2495108>
9. Sharon O, Alpert Y. MAC level Throughput comparison: 802.11ac vs. 802.11n. *Physical Communication*. 2014; 12:33–49. <https://doi.org/10.1016/j.phycom.2014.01.007>
10. Kim BS, Hwang HY, Sung DK. Effect of Frame Aggregation on the Throughput Performance of IEEE 802.11n. *IEEE Wireless Communications and Networking Conference (WCNC)*; 2008.
11. Sharon O, Alpert Y. The combination of QoS, aggregation and RTS/CTS in Very High Throughput IEEE 802.11ac networks. *Physical Communication*. 2015; 15:25–45. <https://doi.org/10.1016/j.phycom.2015.01.003>
12. Li T, Logothetis D, Veeraraghavan M. Analysis of a Polling System for Telephony Traffic with Application to Wireless LANs. *IEEE Transactions on Wireless Communications*. 2006; 5(6):1284–1293. <https://doi.org/10.1109/TWC.2006.1638649>
13. Hu J, Min G, Woodward ME. Modelling of IEEE 802.11e Contention Free Bursting Scheme with Non-Identical Stations. *IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems (MASCOTS)*; 2007.
14. Zhao Q, Tsang DHK, Sakurai T. Modeling Nonsaturated IEEE 802.11 DCF Networks Utilizing an Arbitrary Buffer Size. *IEEE Transactions on Mobile Computing*. 2011; 10(9):1248–1263. <https://doi.org/10.1109/TMC.2010.258>
15. van Eenennaam M, Hendriks L, Karagiannis G, Heijenk G. Oldest Packet Drop (OPD): a Buffering Mechanism for Beaconing in IEEE 802.11p VANETs. *IEEE Vehicular Networking Conference (VNC)*; 2011.
16. Sundaresan K, Anantharaman V, Hsieh HY, Sivakumar R. ATP: A Reliable Transport Protocol for Ad Hoc Networks. *IEEE Transactions on Mobile Computing*. 2005; 4(6):588–603. <https://doi.org/10.1109/TMC.2005.81>
17. Obaidat I, Alsmirat M, Jararweh Y. Completing IEEE 802.11e Implementation in NS-3. *International Conference on Information and Communication Systems (ICICS)*; 2016.
18. Network Simulator ns-3; 2017. Available from: <https://www.nsnam.org>.
19. Kim S, Cho YJ. Adaptive Transmission Opportunity Scheme Based on Delay Bound and Network Load in IEEE 802.11e Wireless LANs. *Journal of Applied Research and Technology*. 2013; 11(4):604–611. [https://doi.org/10.1016/S1665-6423\(13\)71568-2](https://doi.org/10.1016/S1665-6423(13)71568-2)
20. Lee JY, Hwang HY, Shin J, Valaee S. Distributed Optimal TXOP Control for Throughput Requirements in IEEE 802.11e Wireless LAN. *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*; 2011.
21. Karmakar R, Swain P, Chattopadhyay S, Chakraborty S. Performance Modeling and Analysis of High Throughput Wireless Media Access with QoS in Noisy Channel for Different Traffic Conditions. *International Conference on Communication Systems and Networks (COMSNETS)*; 2016.
22. Jin H, Jung BC, Sung DK. A Tradeoff Between Single-User and Multi-User MIMO Schemes in Multi-Rate Uplink WLANs. *IEEE Transactions on Wireless Communications*. 2011; 10(10):3332–3342. <https://doi.org/10.1109/TWC.2011.080311.101980>
23. Xu K, Tipper D, Qian Y, Krishnamurthy P, Tipmongkongsilpg S. Time-Varying Performance Analysis of Multihop Wireless Networks With CBR Traffic. *IEEE Transactions on Vehicular Technology*. 2014; 63(7):3397–3409. <https://doi.org/10.1109/TVT.2013.2297382>
24. Raptis P, Vitsas V, Paparrizos K, Chatzimisios P, Boucouvalas AC. Packet Delay Distribution of the IEEE 802.11 Distributed Coordination Function. *IEEE Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*; 2005.