Adaptive Packet Scheduling Scheme Based on Network Conditions in WLAN Using Fuzzy System

Bakeel Maqhat, Mohd Dani Baba, Ruhani Ab Rahman, Anuar Saif, Md Mahfudz Md Zan, Mat Ikram, Mohd Asri Mansor and Farok Hj Azmat

Abstract The IEEE 802.11n standard has introduced frame aggregation technique which can reduce the overhead and increases the channel utilization efficiency. However, the standard does not address the packet scheduling during the aggregation, and the aggregation scheme causes additional delays, particularly when waiting for other packets in the queue to construct the aggregated frame. Furthermore, the channel status should be addressed to improve the system performance. In this paper, we proposed a scheduling scheme called Dynamic Sensing Mechanism that handles the influence of network channel conditions during the transmission process. The mechanism decreases the number of expired packets during the retransmission by allowing the packets to be transmitted at an earlier time before their expiration time. The simulation results show an outstanding performance improvement for the proposed scheduling mechanism by reducing the packet loss ratio and increasing the system throughput.

Keywords 802.11n · Channel state · Fuzzy logic system · WLAN scheduling · QoS

1 Introduction

With the increasing demands for multimedia applications in wireless LAN systems, it has become essential to provide services with enhanced quality of service (QoS). The IEEE 802.11n standard is introduced to achieve more than 100 Mbps of throughput at the MAC layer and to enhance the QoS requirements [1].

B. Maqhat (✉) · M.D. Baba · R.A. Rahman · A. Saif · M.M.M. Zan · M. Ikram
M.A. Mansor · F.H. Azmat
Centre for Computer Engineering Studies, Faculty of Electrical Engineering,
Universiti Teknologi MARA, 40450 Shah Alam, Selangor, Malaysia
e-mail: bakeelyem@gmail.com

M.D. Baba
e-mail: mdani074@salam.uitm.edu.my

P.J. Soh et al. (eds.), *Theory and Applications of Applied Electromagnetics*,
Lecture Notes in Electrical Engineering 379, DOI 10.1007/978-3-319-30117-4_3

However IEEE 802.11 standard does not specify a scheduling algorithm, its left for the vendors to interpret. The current 802.11n scheduler inherits the priority mechanism of the legacy 802.11e Enhanced Distributed Channel Access (EDCA) scheduler. EDCA scheduler defines several access categories (ACs). A number of studies have been tried to implement scheduling mechanism for WLAN.

Wang and Zhuang [2] have proposed a novel token-based scheme to eliminate collisions and subsequently increases channel utilization. The authors have reported that by integrating voice and data traffic, the token-based scheduling can lead to a better result than DCF in terms of channel utilization. Inan et al. [3] introduced an application-aware adaptive HCCA scheduler for IEEE 802.11e WLANs. This scheduling algorithm is based on the Earliest Deadline First (EDF) scheduling discipline to make polling order based on the computed deadlines of the traffic. According to traffic specification along with instantaneous buffer occupancy information in time sensitive application, this algorithm schedules multimedia traffic by associating each QoS station with a distinct service interval (SI) and TXOP. However, the drawback is the additional hardware/firmware complexity that imposed by the algorithm which makes it not easy to implement. Frantti [4] introduces a fuzzy expert system to adapt the packet size for VoIP traffic in ad hoc networks. The model requires two variables input for the fuzzy system which are packet error rate and change of packet error rate. The results show that the expert system is capable to locate packet size values to the optimum level quickly along with the increment in the number of VoIP connections. Alsahag et al. [5] proposed a bandwidth allocation algorithm for the uplink traffic in mobile WiMAX called FADDR. The algorithm uses fuzzy logic control which is embedded in the scheduler with adaptive deadline-based scheme to guarantee a particular maximum latency for real-time traffics and maintain the minimum requirements for the non-real-time traffics. Seytnazarov and Kim [6] have introduced QoS-aware A-MPDU scheduler, which is applied to voice real-time traffic by controlling the delay time of whole buffering for A-MPDU and configure the buffering separately depending on access category and IP address for the destination. Ramaswamy et al. [7] have proposed Bi-Scheduler algorithm which separates frames depending on their access categories and schedules the VoIP traffic using A-MSDU aggregation, while scheduling the video and non-real-time traffic using the A-MPDU aggregation. Nevertheless, the algorithm is not effective under low-traffic load and the high sensitive traffic. Moreover, it may suffer from delay due to waiting for transmitting. The frame aggregation scheduler proposed by Selvam and Srikanth [8] have crystallized the approach of aggregate frames by calculating the deadline based on the earliest expiry time of a frame waiting in the queue and selecting dynamically the aggregation scheme based on frame aggregation size and bit error rate by using optimal frame size from the lookup table. However, this algorithm is restricted to deal with one type of traffic; thus, other traffic will suffer from delay and eventually may affect the QoS requirements of other traffics. The works in [9–11] have proposed schedulers by exploiting the A-MSDU attributes to enhance the system performance. Maqhat et al. [10] proposed a real time scheduling algorithm for A-MSDU aggregation (RSA-MSDU), which schedules the traffics that are time sensitive and have different lifetimes. The work requires a queue for the sender side, that is, the transmitting queue (TQ). In the TQ, the MSDUs will be sorted in ascending order based on their priorities. According to that, packet with the highest priorities will occupy the top of the queue. During aggregation, the packet is taken from the top of the TQ and attached with the aggregation headers. Then, the packet (subframe) is put into the aggregated frame which is referred as the superframe. The TQ sorting process is updated whenever a new packet is added to the TQ or an acknowledgment bitmap (ACK) is received. During the construction of the aggregation frame, only the packets that have the same destination address (DA) will be associated with the aggregation frame. The aggregated frame is transmitted as soon as it reaches the aggregation size limit or the aggregation delay limit (T_{agg}). The aggregation delay limit is estimated based on arrival time of the head packet in the queue (H_{arv}), the current time (T_{crt}), and the absolute time required to transmit an aggregated frame and receive its block acknowledgment (T_{tx}). The absolute time limit to transmit an aggregated frame is the T_{agg} without compromising the delay of the aggregated packets. The T_{agg} can be calculated by the following equation:

$$T_{\mathrm{agg}} = (T_{\mathrm{crt}} - H_{\mathrm{arv}})/T_{\mathrm{tx}} \tag{1}$$

where T_{agg} is the aggregation delay limit, H_{arv} is the arrival time of the head packet in the queue, and T_{crt} is the current time.

The TQ is updated once a Block Acknowledgment is received. When packet is received correctly, it will be removed from the TQ; otherwise, it will be considered as corrupted. The corrupted packet will gain a high sending priority by placing it at the top of TQ and will be retransmitted at the head of the next aggregated superframe. The drawback of this scheme is that sending of the superframe is done based on the time required for transmitting the superframe, without taking into consideration the lifetime of the retransmitted packets. Ignoring such factor will result in removing the dead packets which in turns affect the packet loss ratio.

Addressed this issue is presented in this paper by introducing a Dynamic Sensing Mechanism (DSM) scheme, which handles the influence of network channel conditions for the transmission process. DSM scheme uses the fuzzy expert system to calculate dynamically the transmission time.

2 Dynamic Sensing Mechanism

Dynamic Sensing Mechanism (DSM) scheme handles the influence of network channel conditions for the transmission process. This is done by investigating the factors that affect the network channel state such as the noise and network load. DSM scheme estimates the traffic load and noise and calculates the time required for successful transmission. The estimation in this mechanism is based on the assumption that by increasing the traffic load and noise will increase the number of failed packets and thus will increase the time required for successful transmission. Hence, the DSM scheme will allow the retransmission attempts before packets expired. The mechanism relies on two sensors, erroneous channel sensor (ECS) and traffic load sensor (TLS), where ECS indicates the amount of failed packets, caused by the noise. TLS indicates the attempts of superframes transmitted which caused by traffic congestion. Moreover, the noise has an effect on the traffic load. Having a high noise in the network causes a bit error in the packet transmission and increases the retransmission attempts. DSM mechanism is able to decrease the number of expired packets during retransmission by allowing the packets to be transmitted at an earlier time before expiration. This behavior will enhance the performance by reducing the packet loss ratio.

To investigate the effect of the network load and noise, we have conducted some experiments in different channel conditions. In these experiments, the number of users varies from 2 to 60 and the channel condition varies from error-free channel to erroneous channel. The results are collected and then analyzed statistically using SPSS program. During the analysis, we used the linear regression between ECS, TLS, traffic load, and noise. We set the noise and traffic load as independent factors and ECS and TLS as dependent factors during the analysis. Results show that traffic load and noise have a significant effect on TLS. Noise and traffic load predict 0.911 of TLS, (0.654 and 0.254) for the two factors, respectively. However, ECS is affected significantly only by noise and not by traffic load. Noise predicts 0.988 of ECS, whereas traffic load predicts only 0.008 that is a negligible value. The linear regression results give an indication to what extent the noise and load affect ECS and TLS. Therefore, we can rely on ECS and TLS in determining the performance of the network.

The DSM allows the packets to be transmitted even in bad network conditions. In this case, the lifetime of the packets will be the criteria for removing the packet from transmitting queue rather than the number of retry attempts.

2.1 Calculate the Transmission Time Using Fuzzy System

Enhance the work by performing adaptive and dynamic mechanism which computes an optimal time for the transmission process better than in static methods.

DSM uses an expert system based on fuzzy logic to enable the scheduler to make a transmission decision based on the network channel conditions. This can be implemented with an embedded system. Fuzzy logic is a concept which helps computers in making decisions in a way which resembles human behaviors.

DSM scheme dynamically updates the weight of which is the amount of time can be added to the aggregation delay limit to allowed packets to be transmitted before expiration. This weight is an embedded fuzzy system output. We aim in this work to ensure that we compute an optimal time for frame transmission decision in each transmission process which take into account packet deadline and channel status. Figure 1 describes the proposed DSM with embedded fuzzy system.

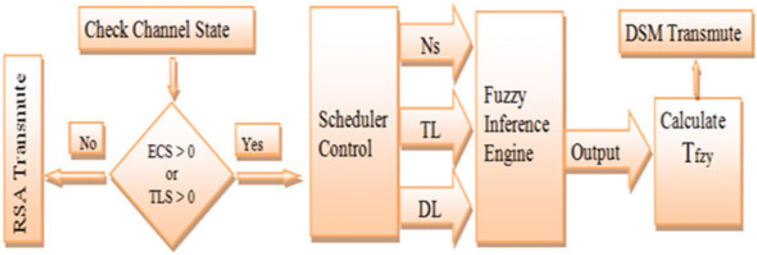


Fig. 1 DSM scheme using embedded fuzzy system

This work develops an embedded fuzzy system to dynamically compute the required time for transmission decision more accurately and with low complexity.

The sensors provide input variables into the system from which decisions can be made, three variables used as the embedded fuzzy system inputs. First is the amount of noise (Ns), the second variable is amount of traffic load (TL), and the third is the head-of-line packet deadline (DL).

2.2 *Fuzzy Reasoning Inference Engine Control Model*

As mentioned earlier, the embedded fuzzy system uses three variables and one variable as inputs and output, respectively. In this system, we design fuzzy logic from fuzzy set theory to design a theoretical structure for the linguistic information where the most important design is to utilize the expert information for the rule base creation. An individual based inference method with Mamdani's design [12] is used, where the inference system rules are jointed into one value. The fuzzy system consists of the following stages: fuzzification, fuzzy reasoning inference, and defuzzification.

The role is to dynamically analyze all input traffics and combine them into one overall fuzzy set. Firstly, the fuzzification process handles three input variables: Ns, TL, and DL, for the overall system. Then, reasoning inference mechanism contains the rule base to manipulate the input variables as shown in Fig. 2. At this point, the actual decision is made representing the human expert process which performs to the linguistic behavior to obtain the output value.

Lastly, the defuzzification phase calculates crisp numerical values to obtain the required weight, which provides an indication of the priority for the scheduler.

Three linguistic levels have been defined for input variables: Ns, TL, and DL, namely low, medium, and high, and five levels for output such as very low, low, medium, high, and very high. As the utilized fuzzy system considers three variables as inputs and three membership functions are considered for each, subsequently the rule base composed of 27 rules (see Table 1). The dynamically normalized scale for the DL input variable is formed from 0 to 200 ms, whereas the dynamically normalized scale for the other inputs and output variables is formed from 0 to 1.

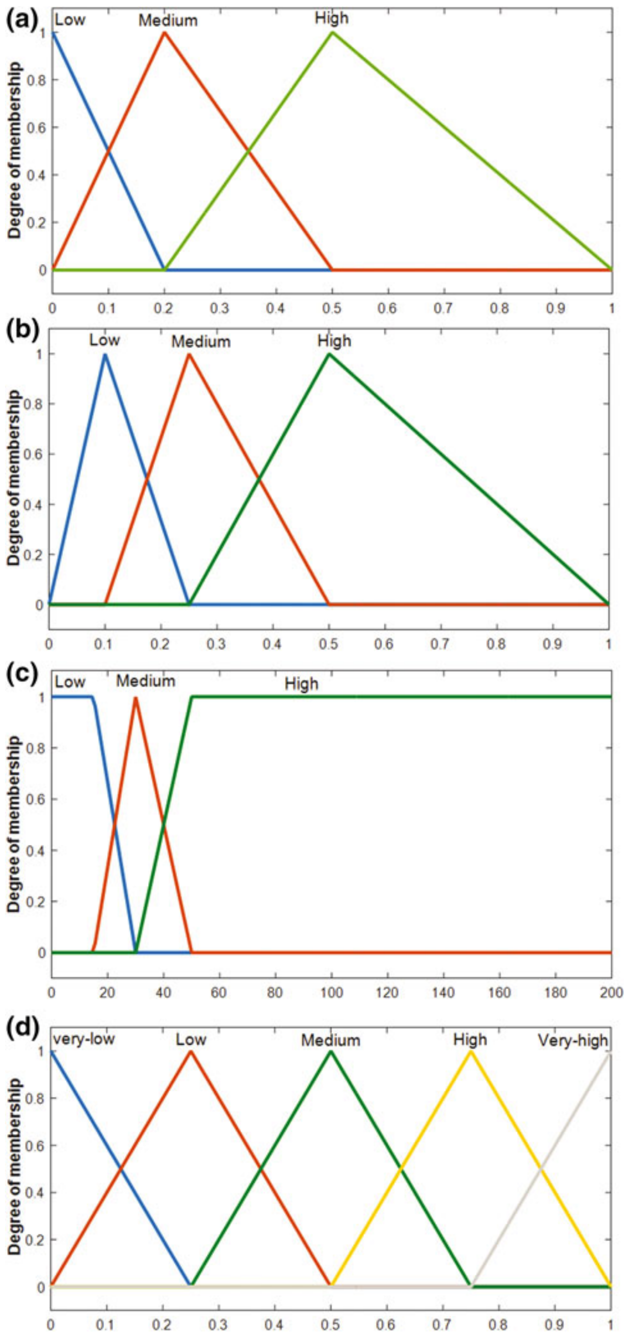


Fig. 2 Membership functions. **a** Noise, **b** traffic load, **c** deadline, **d** output w_{tf}

Table 1 Fuzzy system rule base

Rules N	Noise	T. Load	Deadline	Output w_{tf}
0	Low	Low	Low	Low
1	Low	Medium	Low	Low
2	Low	High	Low	High
3	Low	Low	Medium	Very low
4	Low	Medium	Medium	Low
5	Low	High	Medium	Medium
6	Low	Low	High	Very low
7	Low	Medium	High	Very low
8	Low	High	High	Low
9	Medium	Low	Low	High
10	Medium	Medium	Low	Very high
11	Medium	High	Low	Very high
12	Medium	Low	Medium	Medium
13	Medium	Medium	Medium	Medium
14	Medium	High	Medium	Medium
15	Medium	Low	High	Low
16	Medium	Medium	High	Low
17	Medium	High	High	Medium
18	High	Low	Low	Very high
19	High	Medium	Low	Very high
20	High	High	Low	Very high
21	High	Low	Medium	Medium
22	High	Medium	Medium	Medium
23	High	High	Medium	High
24	High	Low	High	Medium
25	High	Medium	High	Medium
26	High	High	High	Medium

Weight value (w_{tf}) obtained from the fuzzy inference control system is the time factor ratio which is used to compute the optimal time that enables the scheduler to make a decision to send the superframe based on the network channel conditions in order to support packets retransmission. The new aggregation delay limit is given by Eqs. 2 and 3.

$$T_{fzy} = T_{tx} + \mathrm{DL} \times w_{tf} \tag{2}$$

$$T_{agg} = \frac{(T_{crt} - H_{arv})}{T_{fzy}} \tag{3}$$

where T_{fzy} is the new transmission time which is computed by fuzzy system. This scheme is capable of decreasing the number of expired packets during retransmission by allowing the packets to be transmitted at an appropriate earlier time before expiration, particularly in the bad channel conditions. This behavior will enhance the QoS requirements by reducing the packet loss ratio and increase the system throughput. The pseudocode of DSM system is presented in Algorithm 1.

2.3 Explain Fuzzy System Work by Examples

The designed fuzzy system will update the weights of the transmission time of IEEE 802.11n scheduler. The primary aim of the system was to adapt itself to variations of network channel conditions by providing an optimal time to make frame transmission decision to allow the packets to be transmitted before expiration.

In this section, as an example to explain this system: suppose the sensors sense the amount of noise and traffic load with the values of 0.27 and 0.14 for noise and traffic load, respectively. And the scheduler specifies the deadline with the value of 19 ms. The fuzzy system will read these variables as its inputs. Considering the membership functions in Fig. 2, we can make out the linguistic values for three input variables of 0.27, 0.14, and 19 and can be read as follows:

Let the amount of noise be 0.27 which after fuzzification is in linguistic form medium at grade membership of 0.8 and high at grade membership of 0.20 (see Fig. 2a). Let the load be 0.14 which is after fuzzification low at grade membership of 0.7 and medium at grade membership of 0.3 (see, Fig. 2b). In addition, the deadline (DL) is 0.19 ms which means it is low at grade membership of 0.65 and medium at grade membership of 0.35 (see, Fig. 2c).

Applying fuzzy reasoning with rule base from Table 1 and Fig. 2, we can read as:

- If the noise is medium (0.8) AND load is low (0.70) AND DL is low (0.65) THEN the weight value (w_{tf}) is high at grade membership of 0.73 [Rule 9].
- If the noise is medium (0.80) AND load is low (0.70) AND DL is medium (0.35) THEN w_{tf} is medium at grade membership of 0.26 [Rule 12].
- If the noise is medium (0.80) AND load is medium (0.30) AND DL is low (0.65) THEN w_{tf} is very high at grade membership of 0.26 [Rule 10].
- If the noise is medium (0.80) AND load is medium (0.30) AND DL is medium (0.35) THEN w_{tf} is medium at grade membership of 0.26 [Rule 13].
- If the noise is high (0.2) AND load is low (0.70) AND DL is low (0.65) THEN w_{tf} is very high at grade membership of 0.23 [Rule 18].
- If the noise is high (0.2) AND load is low (0.70) AND DL is medium (0.35) THEN w_{tf} is medium at grade membership of 0.23 [Rule 21].
- If the noise is high (0.2) AND load is medium (0.30) AND DL is low (0.65) THEN w_{tf} is very high at grade membership of 0.23 [Rule 19].
- If the noise is high (0.2) AND load is medium (0.30) AND DL is medium (0.35) THEN w_{tf}. is medium at grade membership of 0.23 [Rule 22].

From the above rules and using Mamdani's inference, we can conclude that the weight value will be high at grade membership of 0.73 and medium at grade

Table 2 Examples for input parameters and their fuzzified values

Inputs variables			Crisp output w_{tf}
Noise	T. load	Deadline	
0.27	0.14	19	0.72
0.04	0.60	8	0.80
0.20	0.10	50	0.25
0.60	0.45	36	0.625
0.80	0.90	60	0.50
0.05	0.26	100	0.085

membership of 0.27. The final value of output weight of time is calculated after defuzzification which gives a crisp value of 0.72. It will be used as time factor to calculate an optimal time for scheduler to transmit a superframe with taking into consideration the bad channel conditions.

For every three input variables, the intelligent fuzzy system generates one crisp value. More examples are shown in Table 2.

Algorithm 1. Pseudo code of DSM algorithm

```
if superframe is not null then
    Check the channel state
    if attempts of superframes > 0 then
        TLS=attempts of superframes/Max(attempts of superframes)
    end if
    if retrylimit > 0 then
        ECS = failed packets /(total packets )
        fuzzy first input= Ns
        fuzzy second input=TL
        fuzzy third input=DL
    end if
    calculate transmission time T_fzy Eq (2)
end if
if channel is idle l then
    send the superframe
end if
```

3 Results and Discussions

We have carried out several simulation experiments using the network simulator (NS2) in order to evaluate the scheduling of time sensitive traffics in terms of packet loss, throughput, and average delay. Moreover, we have used the simulation scenarios number 17 of the point-to-point usage model [13]. The scenario consists of a single-hop WLAN in which the transmission power of all the high-throughput STAs is high enough to ensure no hidden terminals in the network. All the stations are operating over a 20 MHz. Furthermore, the network traffic is composed of VoIP with a packet size of 120 bytes, video conferencing with a packet size of 512 bytes, and Internet video/audio streaming with a packet size of 512 bytes. The VoIP rate is 0.96 Mbps, video conferencing rate is 2 Mbps, and the Internet streaming video/audio rate is 2 Mbps. The characteristic of traffic is given in Table 3.

The data rates are set to 150 and 300 Mbps and basic rate is set to 54 Mbps. The average delay, throughput, and packet loss ratio of the different traffic are examined under a different number of stations (i.e., varied from 10 to 60) and different data rates with a bit error rate (BER) is set to 10^{-4}. Other simulation parameters are listed in Table 4.

The average delay in the DSM scheme and RSA-MSDU is shown in Figs. 3, 4, and 5. The proposed DSM scheme achieves smaller average delay due to its ability to transmit the packets at a sufficient time before their expiration time; thus, the

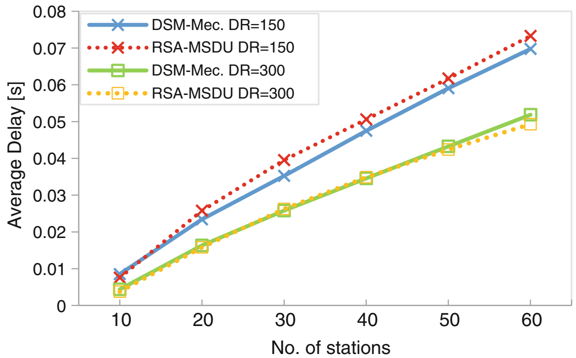
Table 3 Application characteristics

Application	Priority	Data rate (Mbps)	Lifetime (ms)
VoIP	Very high	0.096	30
Video conferencing	High	2	100
Internet streaming	Medium	2	200

Table 4 Simulation parameters

Parameter	Value
T_{SIFS}	16 μs
T_{PHYhdr}	20 μs
T_{idle}	9 μs
CW_{min}	16
T_{DIFS}	34 μs
Basic rate	54 Mbps

Fig. 3 VoIP average delay



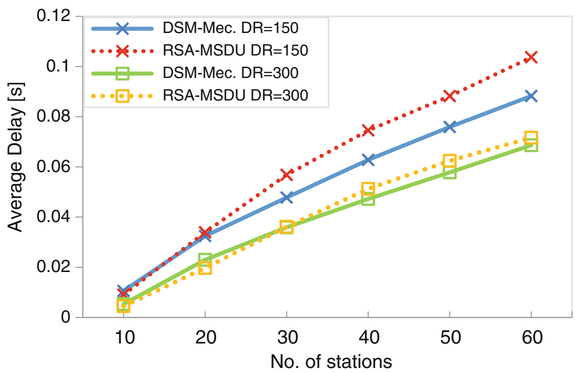


Fig. 4 Video conferencing average delay

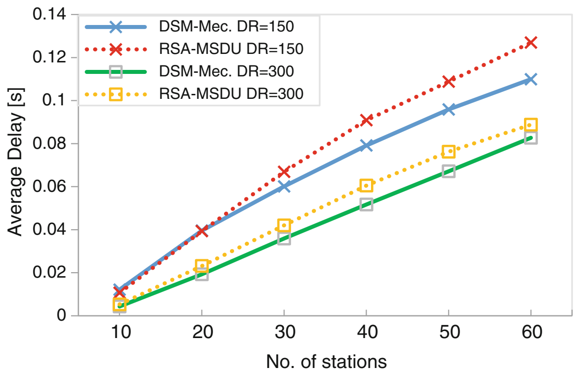


Fig. 5 Internet streaming video/audio average delay

packets will not suffer a long queuing delay. The lower the contending station, the smaller the average delay as the stations will have frequent access to the medium and the packets will not suffering a long queuing delay. The performance gain for the DSM scheme over the RSA-MSDU is about 6 % for VoIP in the case of 150 Mbps and no gain is recorded under 300 Mbps data rate where packets will not suffer from long queuing delay due to the high data rate. For video conferencing, the gain is about 14 % in the case of 150 Mbps, while it is about 6 % at high data rates. Even with the high delay tolerance in video conferencing, DSM still able to improve the performance. The DSM enhancement for Internet streaming video/audio is about 12 % for both data rates.

The packet loss ratio of DSM mechanism compared to RSA-MSDU is shown in Figs. 6, 7, and 8. With a small number of competing network stations, the packet loss will be small because the network will have a low number of superframes compete to transmit. The packet loss will rise with the increasing number of stations. Moreover, the figures show the outstanding performance of our mechanism in reducing the packet loss ratio at the high noise and the high traffic load. Figure 6 shows an enhancement in VoIP packet loss of about 30 % for both data rates, while Fig. 7 shows the video conferencing enhancement which is about 52 and 65 % for both data rates.

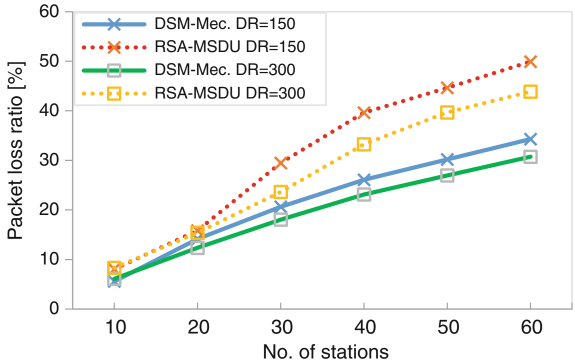


Fig. 6 VoIP packet loss ratio

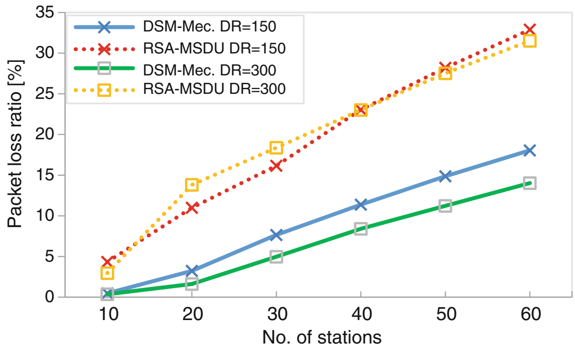


Fig. 7 Video conferencing packet loss ratio

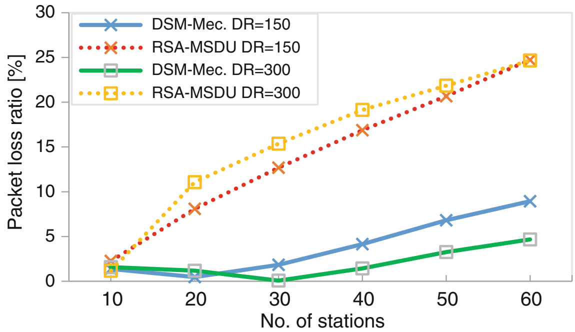


Fig. 8 Internet streaming video/audio packet loss ratio

Internet streaming video/audio follows the same behavior as the other traffics and scores an enhancement of 72 and 87 % under 150 and 300 Mbps, respectively, see Fig. 8. The previous results match with our assumption that the DSM mechanism will have a great impact on reducing the packet loss. Moreover, the results show that whenever the delay bound of the traffic is high, the better the DSM performance. The packet loss enhancement is increased whenever increased the lifetime of the traffics.

Figures 9, 10, and 11 show the throughput performance of the proposed DSM scheduler scheme under a different number of stations. In a large number of stations, the collisions occur repeatedly and impact the system performance. The system throughput keeps decreasing while the network stations increase. Nevertheless, the system throughput of the DSM mechanism reaches about 38 and 53 Mbps, whereas the RSA-MSDU approaches 32 and 47 Mbps at high traffic load under 150 and 300 Mbps, respectively, see Fig. 9. The throughput gain reaches more than 8 % for both video conferencing and Internet streaming video/audio under 150 Mbps. In a data rate of 300 Mbps, the throughput gain reaches 5 and 6 % for video conferencing and Internet streaming video/audio, respectively, see Figs. 10 and 11. The increase in the throughput is attributed to the enhancement of the system efficiency, due to reducing the amount of loss.

Fig. 9 System throughput

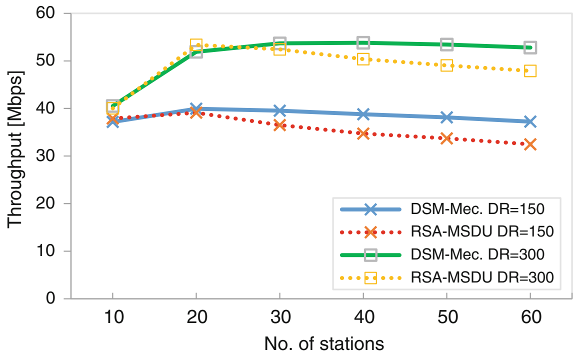
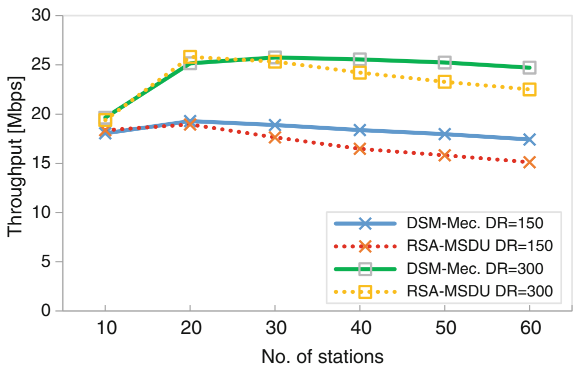


Fig. 10 Video conferencing throughput



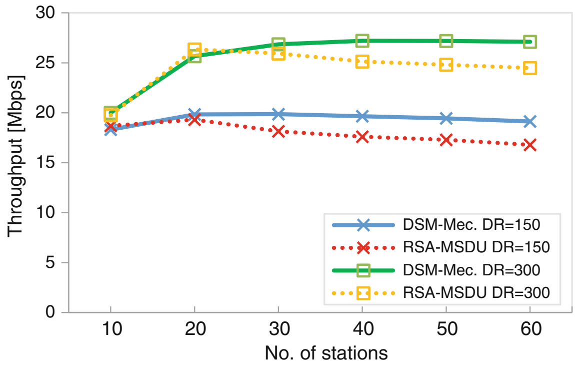


Fig. 11 Internet streaming video/audio throughput

4 Conclusion

In this work, we have introduced a scheduling mechanism called DSM scheme which handles the influence of network channel conditions for the transmission process by estimating the traffic load and noise and then calculates the required time for successful transmission. DSM mechanism uses two sensors: ECS and TLS to sense the amount of noise and traffic congestion in the network. Accordingly, DSM scheme employs the fuzzy expert system to dynamically compute the superframe time transmission. The simulation results show that DSM scheme can significantly improve the system performance by reducing the packet loss ratio to about 80 % and increasing the system throughput.

Acknowledgments The authors would like to acknowledge the Research Management Institute of Universiti Teknologi MARA for providing the research grant for the Project No: 600-RMI/DANA 5/3/REI (2/2015).

References

1. Xiao Y (2005) Performance analysis of priority schemes for IEEE 802.11 and IEEE 802.11 e wireless LANs. IEEE Trans Wireless Commun 4(4):1506–1515
2. Wang P, Zhuang W (2008) A token-based scheduling scheme for WLANs supporting voice/data traffic and its performance analysis. IEEE Trans Wireless Commun 7(5):1708–1718
3. Inan I, Keceli F, Ayanoglu E An adaptive multimedia QoS scheduler for 802.11 e wireless LANs, pp 5263–5270
4. Frantti T Fuzzy packet size optimization for delay sensitive traffic in ad hoc networks, pp 2633–2637
5. Alsahag AM, Ali BM, Noordin NK, Mohamad H (2014) Fair uplink bandwidth allocation and latency guarantee for mobile WiMAX using fuzzy adaptive deficit round robin. J Netw Comput Appl 39:17–25
6. Seytnazarov S, Kim Y-T QoS-aware MPDU aggregation of IEEE 802.11 n WLANs for VoIP services, pp 2–4

7. Ramaswamy V, Sivarasu A, Sridharan B, Venkatesh H (2014) A bi-scheduler algorithm for frame aggregation in IEEE 802.11 n. arXiv preprint arXiv:1401.2056
8. Selvam T, Srikanth S A frame aggregation scheduler for IEEE 802.11 n, pp 1–5
9. Maqhat B, Baba MD, Rahman RA A-MSDU real time traffic scheduler for IEEE802. 11n WLANs, pp 286–290
10. Maqhat B, Baba MD, Ab Rahman R, Saif A (2014) Scheduler algorithm for IEEE802. 11n wireless LANs. Int J Future Comput Commun 3(4):222–226
11. Maqhat B, Dani Baba M, Rahman RA, Saif A Performance analysis of fair scheduler for A-MSDU aggregation in IEEE802. 11n wireless networks, pp 60–65
12. King PJ, Mamdani EH (1977) The application of fuzzy control systems to industrial processes. Automatica 13(3):235–242
13. Stephens AP et al (2004) Technical report, IEEE 802.11n working document 802.11-03/802r23