

IMPROVING QoS IN WLAN USING DYNAMIC WEIGHTED FAIR SCHEDULING

Ahmed Riza¹, Ling Teck Chaw¹, Phang Keat Keong¹, Hoh Kian Pee²

¹Department of Computer System & Technology,
Faculty of Computer Science and Information Technology,
University of Malaya,
50603 Kuala Lumpur, Malaysia.

²School of Electronics & Info-Comm Technology,
ITE College East, Singapore 486047

E-mail: ahmedriza@gmail.com, {tchaw, kkphang}@um.edu.my, hoh_kian_pee@ite.edu.sg

ABSTRACT

The demand for end-to-end Quality of Service (QoS) in Wireless Local Area Networks (WLANs) is becoming ever more important due to the trend towards converged communication networks. The recently accepted QoS mechanism in WLAN: the IEEE 802.11e has limitations in providing fairness to different types of traffic. We proposed Dynamic Weighted Fair Scheduling Scheme (DWFSS) to improve fairness in WLANs by enhancing the IEEE 802.11e. The DWFSS allocates bandwidth to four different classes of traffic. These classes are assigned weights which are dynamically changed. The service interval in IEEE 802.11e is divided into two intervals i.e., High Priority Polling Interval (HPPI) and Low Priority Polling Interval (LPPI). The high priority traffic is polled during HPPI and low priority traffic is polled during LPPI. The performance of DWFSS is evaluated on NCTUns simulator using multiple types of traffic and compared with the standard scheduler. The simulation results showed that DWFSS improved fairness when compared to IEEE 802.11e standard scheduler. DWFSS increased average throughput by 13.48% and reduced average jitter by 16.21% while maintaining an acceptable level of end-to-end delay.

Keywords: Quality of Service, Wireless Local Area Networks, IEEE 802.11e, Fair Scheduling Scheme

1.0 INTRODUCTION

The first Wireless Local Area Network (WLAN) was introduced in the late 1990s [1]. Since then the popularity of WLANs is growing due to improvement in transmission speed, flexibility and low cost infrastructure. With the telecommunication industry trending towards converged networks the demand for multimedia traffic such as video streaming, teleconferencing, and Voice over Internet Protocol (VoIP) on WLANs has increased. As a result, the emphasis for QoS guarantees on delay and jitter sensitive applications in WLAN have increased. This demand is further compounded by the rapid growth in WLAN capabilities being integrated into devices and peripherals [2].

Providing fairness to applications including delay and jitter sensitive applications is vital for the future growth of wireless networks. However, providing a reliable QoS to delay and jitter sensitive applications while maintaining an acceptable level of fairness encounters several challenging task. For traditional wired networks, the primary challenge is congestion [3]. In a converged network when packets take different paths with different levels of congestion or different number of hops, jitter and delay becomes unpredictable. In order to solve these problems and provide QoS in traditional wired networks several schemes have been proposed. Some schemes reserve resources on a per-flow basis and the other on a per-class basis. The most well known mechanisms in Internet Protocol (IP) networks are Integrated Services (IntServ) [4] and Differentiated Services (DiffServ) [5].

Compared to wired networks, WLANs face additional challenges such as channel fading, interference, high propagation delay, and mobility of stations. Hence the traditional wired QoS mechanism cannot be directly applied to the WLANs. In order to provide QoS in WLANs the IEEE 802.11 Task Group E introduced the IEEE 802.11e standard [6]. The 802.11e defines two channel access mechanisms i.e., the Enhanced Distributed Channel Access (EDCA) and the HCF Controlled Channel Access (HCCA). The EDCA is a distributed contention based channel access mechanism. The HCCA is a contention free, channel access mechanism. However, many limitations have been identified in both EDCA and HCCA. When the network is congested EDCA leads to the starvation of low priority traffic [7]. The HCCA

Admission Control Unit (ACU) performs admissions based on First Come First Serve (FCFS) algorithm. Therefore, the standard QoS mechanism, namely 802.11e, does not provide a differentiated prioritized scheduling mechanism and resulted in unfairness and starvation of traffic [8]. This suggested the requirement of an efficient, dynamic and fair bandwidth allocation and admission control algorithm for WLAN.

The remainder of this paper is organized as follows. Section 2 discussed related works. The 802.11e HCCA standard scheduler (SS) and related scheduling algorithms which improve the performance of WLAN are reviewed in Section 3. A new scheduling algorithm is proposed in Section 4, which dynamically allocates bandwidth to different classes. The simulation results are discussed and evaluated in Section 5 in terms of fairness, end-to-end delay, throughput and jitter. The conclusion and possible future works are discussed in Section 6.

2.0 BACKGROUND AND RELATED WORK

This section provides an overview of the IEEE 802.11e HCCA method and some related scheduling algorithms proposed to improve QoS in WLAN is described.

2.1 Background

The 802.11e defines a new coordination function called Hybrid Coordination Function (HCF) which is only usable in a quality of service Basic Service Set (QBSS). The HCF alternates between two access mechanisms i.e., Enhanced Distributed Channel Access (EDCA) and HCF Controlled Channel Access (HCCA). The EDCA is a contention based channel access mechanism and HCCA is a contention free channel access mechanism. Additionally, the 802.11e introduced the concept of Transmission Opportunity (TXOP) in the HCF. The TXOP is an interval of time in which the QoS-aware Station (QSTA) has the right to send multiple frames. Every TXOP is described by a starting time and a maximum duration called TXOP Limit. The QoS Access Point (QAP) sets the TXOP Limit and QSTA transmissions shall not exceed TXOP Limit.

The HCCA is a centralized access mechanism defined for parameterized QoS. It uses a polling-based protocol which is controlled by a QoS-aware hybrid coordinator (HC). Similar to the 802.11, the superframe duration in 802.11e is divided into Contention Free Period (CFP) and Contention Period (CP). Unlike the Point Coordination Function (PCF) in 802.11 [1], the HCCA allows a Controlled Access Phase (CAP) to be initiated at anytime during a CP if the channel remains idle for a time greater than or equal to a PCF Interframe Space (PIFS). This gives HC higher priority to access wireless medium (WM) over other QSTAs.

The HCCA defines the concept of Traffic Class (TC) and Traffic Streams (TS) which allows the HC to provide per-session service for QoS unlike the per-station service in EDCA. The HC grants TXOPs to QSTAs based on the individual QoS requirements of a TS which is called Traffic Specification (TSPEC). The TSPEC structure elements are illustrated in Fig. 1. To establish a TS between the QSTA and the QAP, the QSTA is required to send the TSPEC parameters to the QAP. The QAP may grant up to 8 parameterized TS to each QSTA. The 802.11e standard [6] defines a reference scheduler to control the channel access of the TS.

Element ID (13)	Length (55)	TS Info	Nominal MSDU Size	Maximum MSDU Size	Minimum Service Interval	Maximum Service Interval	Service Start Time
Minimum Data Rate	Mean Data Rate	Peak Data Rate	Burst Size	Delay Bound	Minimum PHY Rate	Suprlus Bandwidth Allowance	Medium Time

Fig. 1 – TSPEC format elements

The SS defined in 802.11e uses a required set of TSPEC elements, to generate a schedule. The minimum TSPEC elements are, Mean Data Rate, Nominal Mac Service Data Unit (MSDU) Size, and Maximum Service Interval or Delay Bound. The HC maintains a polling list of QSTAs requesting for QoS. The scheduler polls all the admitted TS in the polling list during every Service Interval (SI). When a new TS needs to initiate data transmission, the QSTA starts negotiating with the QAP by sending an Add Traffic Streams (ADDTS) request frame which contains the TSPEC

parameters. The HC calculates TXOP duration for the TS, based on the TSPEC information received from the QSTA, and sends an ADDTS response frame to the QSTA. If the HC cannot guarantee QoS for the TS and the TSPEC negotiations failed, the HC rejects the TS and then, informs the QSTA by sending an ADDTS response frame.

When the HC receives an ADDTS request frame the HC computes the scheduled, estimated number of packets for the TS and the corresponding TXOP. To calculate a schedule for admitted streams the scheduler first calculates the scheduled SI. Subsequently, it calculates the TXOP for the TS. According to the SS [6], to calculate the scheduled SI, the scheduler first calculates m , which is the minimum of all the Maximum Service Intervals (MSI) of the admitted flows. Then the scheduler calculates the scheduled SI which is a multiple of the Beacon Interval. This calculated SI should be lower than m . In order to calculate the TXOP duration for the flow i the scheduler first calculate- N_i as in equation (1).

$$N_i = \left\lceil \frac{SI \times \rho_i}{L_i} \right\rceil \quad (1)$$

Where N_i is the number of MSDUs arriving in SI at mean data rate, SI is the scheduled service interval calculated above, ρ_i is the mean data rate and L_i is the nominal MSDU size

Then the scheduler calculates the TXOP duration for flow i as in (2).

$$TXOP_i = \max\left(\frac{N_i \times L_i}{R_i} + O, \quad \frac{M}{R_i} + O\right) \quad (2)$$

Where R_i is the physical transmission rate, M is the maximum allowable size of MSDU (i.e 2304 bytes) and O is the Overheads in time units

Fig. 2 illustrates an example of admitted streams from QSTA i to k . The beacon Interval is 300 ms and the minimum of the MSIs of the three admitted streams is 160ms. The calculated scheduled SI is 150 ms. The scheduler repeats the above mentioned process when a new TS is admitted and recalculates the scheduled SI. If a stream is removed the scheduler reclaims the duration of TXOP and resumes contention.

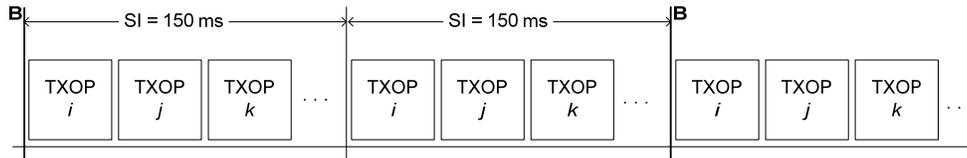


Fig. 2 – Schedule streams from QSTA i to k

As the duration of the scheduled SI is limited, the number of TXOPs granted to QSTAs is limited as well. Therefore, the scheduler is required to implement admission control to ensure QoS for all admitted flows. When a new request for admission is received, the ACU first calculates the scheduled SI and TXOP value as described earlier. Thereafter, the ACU admits the new request if the following inequality is satisfied.

$$\frac{TXOP_{k+1}}{SI} + \sum_{i=1}^k \frac{TXOP_i}{SI} \leq \frac{T - T_{cp}}{T} \quad (3)$$

Where k represents the number of admitted steams in the polling list, $k+1$ is the index used for a newly requesting stream, T is the Beacon interval duration, T_{cp} indicates the CP duration used for contention based EDCA traffic.

2.2 Related Scheduling Schemes

In order to improve the performance of SS, several scheduling schemes have been proposed by various researchers. Venkatakrisnan & Selvakennedy [8] proposed eHCF to improve the ACU. The authors identified that bursty traffic

received considerably low quality of service from both EDCA and HCF and are very likely to starve in large and busy networks. In order to allocate the HC resources each traffic type, traffic type is assigned a percentage of the total bandwidth (3). A Class-Based Queuing (CBQ) mechanism was introduced to provide fairness in WLAN. In CBQ the maximum bandwidth allocated to a class is fixed and it is not possible for any class to acquire bandwidth more than the maximum allocated bandwidth even if there is unused bandwidth in other classes. This admission control method improves performance without fully utilizing the bandwidth.

Gao *et al.* [9] proposed PRBAC which improves the ACU in HCF by taking in to account the channel characteristics and the mobility of stations. Using PRBAC, more TS can be admitted than SS and this increases the number of TXOPs. However, the admission control scheme is prone to packet loss since it uses the average physical rate for admission control. The authors proposed to drop packets from some QoS sessions randomly. However, randomly removing a TXOP for QoS session is not efficient. Didi *et al.* [10] proposed an admission control algorithm considering the load, number of best effort QSTAs and positions of QSTAs. Instead of the minimum transmission rate, the current rate of transmission and the position of the QSTA are used to calculate the TXOP. The authors' simulation tests of this model showed that the algorithm performed better in terms of throughput, latency and collision compared to HCCA. Using this algorithm, if the QSTAs are moving and the QAP needs to calculate the position and rate of QSTAs regularly then the overhead is likely to increase significantly.

Bin Muhamad Noh *et al.* [11,12] proposed two methods, static scheduling scheme and multimedia priority dynamic scheduling (MPDS) scheme. The static scheduling scheme and MPDS scheme were compared with standard scheduler with respect to user level QoS and application level QoS. By using simulation the authors have proved that the static scheduling scheme can maintain a high user level QoS. Additionally results also showed that MPDS performs better than static scheduling scheme and standard scheduler if there are less multimedia QSTAs. The performances of these two schemes were not tested in the presence of other traffic such as data or bursty traffic. Skyrianoglou *et al.* [13] proposed Adaptive Resource Reservation Over WLANs (ARROW). This method uses queue size sub field introduced in 802.11e, to allocate TXOP based on real-time requirements requested by QSTAs. By using simulations, with voice and video traffic, the authors showed that ARROW improves channel utilization and throughput of multimedia traffic. The dynamic TXOP estimation method increases overhead. If the TXOP duration is increased for particular TS during congestion, other flows could experience a delay because the bandwidth agreed for other flows during admission are not offered.

Chen & Yeh [14] proposed ATSP in which the HC calculates a precise time to poll QSTAs which have data ready to be sent, and polls those TS exactly at that time. When a traffic stream is in silent state, the polling interval of the TS is increased up to 300 ms to reduce the overhead. This ATSP scheme can improve throughput, delay and jitter at the expense of greater complexity of the scheduler because the HC needs to calculate the polling time for each TS whenever there is a change in the polling list. In ATSP, the overhead on HC will increase due to the short interval polling period and continuous monitoring of the status of TSs. Lagkas *et al.* [15] proposed POAC-QG in which the beacon interval is divided into two periods, real-time traffic (RT) period and background traffic (BT) period. The POAC-QG is not a polling based access mechanism but a TDMA based scheme. In the admission control procedure, if the bandwidth is enough to serve with maximum quality level, then the algorithm accepts TSs starting with the highest priority TS. When the available bandwidth is not enough to admit a flow with minimum quality, the quality level of previously processed higher priority flows are lowered so that some bandwidth will be saved for the newly arriving flows. The authors showed that the POAC-QG can improve delay and jitter for voice and video traffic compared to HCCA. POAC-QG is an alternative protocol for HCCA and support multimedia traffic only.

Gozalvez *et al.* [16] proposed a scheduling mechanism named hybrid HCCA-EDCA centralized scheme (HHE-CS). All Best Effort (BE) traffic is admitted without any control but the real time traffic is given higher priority over BE traffic in the downlink scheduler. The BE traffic not transmitted during the current CFP period and all uplink BE traffic are served in the next CP through EDCA. With experimental results the authors showed that the HHE-CS scheme can improve the throughput of web traffic without interrupting quality of real time traffic. However, there is a need for extra bandwidth allocated for Variable Bit Rate (VBR) and without having information of the actual required bandwidth from QSTAs makes inefficient use of the bandwidth. Kuo [17] proposed MBDTXOP in which TXOPs are adjusted dynamically according to periodic measurements of assigned bandwidth, unused bandwidth, and TXOP requested. By using simulation results with MPEG videos the author proved that it is more accurate and efficient in providing QoS for VBR video compared to the SS. MBDTXOP dynamically allocated bandwidth to VBR video traffic without considering the

fairness among different traffic classes. In [18] Kim & Cho proposed feedback-assisted scheduling (FAS). Two new frame types CF-M-Poll and mini-frame were introduced. The HC allocates channel time according to a pre-assigned priority of each MPEG frame type. To test the performance of the FAS scheme the authors used simulations which proved that the FAS scheme can improve the channel utilization for MPEG traffic while maintaining a low delay bound. However, the FAS scheme increases the complexity of scheduler. The channel time available for Data frames are also reduced due to the new frames introduced in FAS scheme. The FAS scheme requires modification to both the HC as well as QSTA. Shi *et al.* [19] proposed an optimized traffic scheduling algorithm to improve the performance of VBR over HCCA. The algorithm assigned TXOPs of variable lengths by utilizing the Queue Size subfield. With simulation results the authors showed that the scheme can improve throughput, delay and packet loss ratio. The simulations were carried out with H.261 video traffic. It is unclear how the proposed scheme performs with multiple traffic types.

Elshafei & Baroudi [20] proposed a wireless fair queuing algorithm based on WFQ-R which was proposed by Kim & Yoon [21]. The authors proposed an algorithm to improve the WFQ-R with window-based retransmission. This scheme calculates the amount of extra resources a flow used during retransmission and later provides compensations to the other flows which used less resource. The amount to be charged on a flow is calculated based on the weight of the flow and extra resources used. Then it distributes this amount to backlogged flows in proportion to their weights. The authors showed that this algorithm can improve throughput and decrease queuing delay while maintaining fairness in the presence of errors. Siris & Stamatakis [22] proposed a dynamic CBWFQ service differentiation mechanism to solve the problem of unfairness due to location dependent channel errors, uplink-downlink unfairness and decreased throughput due to low rate stations. Two classes were used, ACK and DATA classes. The unutilized bandwidths of nodes are distributed in proportion to the weight of the classes. Experimental results of this model proved that it can provide fairness in an error channel, resolve uplink-downlink unfairness and reduce the performance anomaly problem. Even though, uplink-downlink fairness problem was addressed in this scheme, there exist a trade-off between fairness and the complexity of the algorithm. Boggia *et al.* [23] proposed a feedback based dynamic bandwidth allocation scheme with a measurement based call admission control algorithm for providing delay guarantees to real-time multimedia flows. The queue lengths from QSTAs were used to dynamically allocate bandwidth to multimedia flows. The HC is required to calculate the bandwidth assigned to each active TS at the beginning of every beacon interval. The algorithm accepts more TXOPs than the SS when the network is overloaded. To avoid exceeding the CAP limit, TXOPs are reallocated by reducing the TXOPs by a fraction of the total sum of TXOPs. However, this reduction violates the agreed services with QSTAs and may decrease QoS provided to multimedia traffic.

3.0 DYNAMIC WEIGHTED FAIR SCHEDULING SCHEME

In this section we present the proposed algorithm to improve QoS in WLAN namely, Dynamic Weighted Fair Scheduling Scheme (DWFSS). We introduce a new dynamic weight adjustment algorithm to provide fairness and increase bandwidth utilization. A new scheduling algorithm which can provide differentiated services to both high and low priority traffic is also presented. Based on the literature review it is understood that the SS [6] is unable to provide bandwidth guarantees. The SS shows poor performance in terms of fairness and is unable to prevent starvation. The main reason for this poor performance is due to the first-come first-serve (FCFS) admission control algorithm [8,24]. In the FCFS mechanism, any flows requesting for bandwidth will be granted the requesting bandwidth until the bandwidth is fully exhausted. Moreover, in FCFS algorithms the bandwidth and medium access rate is only determined by the order of arrival. When the bandwidth is exhausted the scheduler starts to reject new flows until some of the admitted flows complete their transmission. Hence this causes unfairness and starvation to the traffic arriving later. Thus, a dynamic bandwidth management mechanism is needed to solve this unfairness problem in WLAN. In order to overcome the unfairness problems faced by the standard scheduler we propose a new algorithm, the DWFSS. The purpose of this algorithm is to provide bandwidth reservation and fairness during admission control. The DWFSS defines functionalities to dynamically allocate bandwidth to different types of traffic. The DWFSS will be able to differentiate traffic based on the priority of traffic and provide services based on the requirements of each flow. DWFSS is also a low overhead algorithm as it uses the same simple algorithm for both static and mobile nodes.

3.1 Bandwidth Allocation

In order to identify an efficient bandwidth allocation model, some of the well known congestion management models are analyzed. Similar to the WFQ algorithm proposed by Demers *et al.* [25] and DWFQ algorithm proposed by Kun *et al.* [26], the DWFSS can be designed to dynamically allocate bandwidth among different traffic classes. The [27] described a class based link sharing model for traditional wired networks. This link sharing model can be used to maximize the bandwidth utilization. However, in 802.11 WLANs, the channel time is pre-allocated for control and management frames. Therefore, the allocation model described in [27] needs to be modified accordingly. The modified bandwidth sharing model with the classification is described in Table 1.

Table 1 - Classification and bandwidth allocation

TSID	Class	Type of Traffic	Initial Weight
8	1	Data (BK)	5%
9	1	Data (BK)	
10	2	Data (BE)	25%
11	2	Data (BE)	
12	3	Video	40%
13	3	Video	
14	4	Voice	30%
15	4	Voice	

The most important component of the bandwidth allocation algorithm is differentiating different types of traffic. The 802.11e introduced Traffic Stream Identification (TSID) to distinguish MAC Service Data Unit (MSDU) for parameterized QoS. The TSID value range is 8 through 15 and is assigned in the layers above the MAC. The classes can be derived by using TSID instead of User Priority (UP) used in deriving EDCA access categories. Each class can be assigned a weight as a percentage of total bandwidth. The class weights can be dynamically changed according to the requirements and available bandwidth. The proposed bandwidth adjustment algorithm is presented in the following subsection.

3.2 Weight Adjustment Algorithm

If the unused bandwidth of the requesting class is not enough to accept the TS then, the additional amount required by the requesting class is calculated as in equation (4).

$$Required = TXOP_{requesting} - Unused_{requesting} \quad (4)$$

Where *Required* is the additional amount of bandwidth needed by the requesting class, $TXOP_{requesting}$ is the TXOP of requesting class and $Unused_{requesting}$ is the unused bandwidth available in requesting class.

The maximum bandwidth a class can offer is a percentage of unused bandwidth in that class, as calculated in equation (5). An offering class shall not offer all the unused bandwidth to another class.

$$Offer_{max} = Unused_{offering} \times Weight_{offering} \quad (5)$$

Where $Offer_{max}$ is maximum bandwidth the offering class can offer, $Unused_{offering}$ is the total unused bandwidth in the offering class; $Weight_{offering}$ is the current weight of the offering class.

It is not necessary to offer an amount more than a requesting class needs even if the offering class has lot of unused bandwidth. Thus, the offering class will offer either the *Required* as calculated in equation (4) or the $Offer_{max}$ as calculated in equation (5), whichever is the minimum as in equation (6).

$$Offer = \min(Offer_{max}, Required) \quad (6)$$

Where the *Offer* is the actual offered bandwidth by an offering class.

Subsequently, the sum of all the offered bandwidth is calculated until the offered bandwidth is more than or equal to the required bandwidth as in equation (7).

$$Offer_{total} = Offer_m + \sum_{i=1}^{m-1} Offer \geq Required \quad (7)$$

Where *Offer_{total}* is the total offered bandwidth from all offering classes, *Offer_m* is the offered bandwidth from the current offering class and *m* represents the number of offering classes and *m* is less than or equal to 3.

If the *Offer_{total}* is not less than the required bandwidth then the class weights are adjusted accordingly. The weight adjusted from both the requesting and offering class is calculated based on the offer, as in equation (8).

$$Weight_{adj} = \frac{Offer}{Bandwidth_{total}} \quad (8)$$

Where *Offer* is the offered bandwidth by the offering class, *Bandwidth_{total}* represents the total bandwidth and *Weight_{adj}* is the weight adjustment.

The weight of an offering class is reduced as in equation (9).

$$Weight_{offering} = Weight_{offering} - Weight_{adj} \quad (9)$$

Where *Weight_{offering}* is the weight of the offering class and *Weight_{adj}* is the weight adjustment calculated in (8).

The weight of the requesting class is increased as shown in equation (10).

$$Weight_{requesting} = Weight_{requesting} + Weight_{adj} \quad (10)$$

Where *Weight_{requesting}* is the weight of the requesting class. It is important to note that if the *Offer_{total}* is less than *Required* then the class weights are not changed. Fig. 3 shows the weight adjustment algorithm.

Due to the dynamic nature of weight adjustment algorithm, theoretically it is possible that the entire bandwidth of a particular class may be consumed by another class thus causing starvation. In order to prevent this possibility it is necessary to use maximum and minimum percentages. Based on the bandwidth allocation model described in [27] maximum and minimum percentages are derived as shown in Table 2.

Table 2 – Minimum and maximum admissible percentage

TSID	Class	Priority	Min. Guaranteed	Max. Allowed
8	1	Low	50%	96%
9	1	Low		
10	2	Low		
11	2	Low		
12	3	High	52%	75%
13	3	High		
14	4	High		
15	4	High		

Algorithm:	WEIGHT_ADJUSTMENT
Input:	Requesting Class ID, txop of requesting TS
Output:	True / False

1. $count \leftarrow 1$; $offeringClass \leftarrow maxUnusedClass$; $totalOffered \leftarrow unusedReq$
2. **while** $totalOffered$ less than $txop$ **and** $count$ less than 4 **in** $offeringClass$ **do**
3. $required \leftarrow txop - totalOffered$
4. $OfferMax \leftarrow unusedOffering \times WeightOffering$
5. $Offer \leftarrow$ minimum of $OfferMax$ and $required$
6. $totalOffered \leftarrow totalOffered + Offer$
7. $weightAdj \leftarrow Offer / bandwidthTotal$
8. $weightOffering \leftarrow weightOffering - weightAdj$
9. $weightRequesting \leftarrow weightRequesting + weightAdj$
10. $offeringClass \leftarrow maxUnusedClass$
11. **Next** $offeringClass$
12. **if** $totalOffered$ less than $txop$ **then**
13. $weightClass1 \leftarrow prevWeightClass1$
14. $weightClass2 \leftarrow prevWeightClass2$
15. $weightClass3 \leftarrow prevWeightClass3$
16. $weightClass4 \leftarrow prevWeightClass4$
17. **return** false

Fig. 3 – Weight adjusting algorithm

3.3 Scheduling Algorithm

According to ITU [28] different applications have different QoS requirements and network performance parameters such as delay, jitter and loss which affect end user Quality of Experience (QoE). By considering the QoE for delay and jitter sensitive applications and data applications, two main types of applications can be identified. They are delay and jitter sensitive real-time applications and delay insensitive data applications. The delay and jitter sensitive applications are considered as high priority traffic and delay insensitive data applications are considered as low priority traffic. As in [28] the delay requirements of data applications are not stringent. All these are taken into consideration in designing the DWFSS scheduler.

In order to provide differentiated services to high priority traffic and data traffic, the DWFSS introduces two polling intervals. The two polling intervals are named as High Priority Polling Interval (HPPI) and Low Priority Polling Interval (LPPI). The HPPI duration is the total channel time allocated to high priority traffic. All the high priority traffic is polled during the HPPI duration. The LPPI is the duration in which low priority traffic is polled. The LPPI is the remaining time in the service interval after the HPPI. During the LPPI, low priority traffic is polled based on the requirements of low priority traffic. Both polling interval durations vary dynamically as the number of high priority traffic flows change. The polling schedule is calculated as in the algorithm described in Fig. 4.

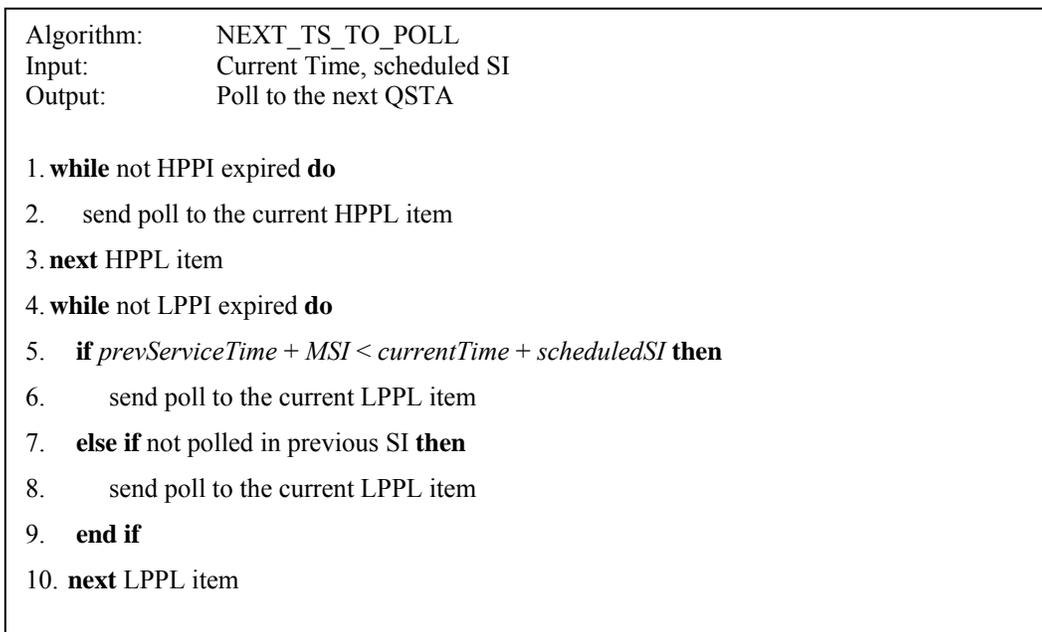


Fig. 4 – Polling Schedule

The DWFSS takes in to account the priority of the TS (based on TSID) and the Maximum Service Interval (MSI) of TS when computing a schedule. Unlike the standard scheduler, DWFSS maintains two polling lists; the High Priority Polling List (HPPL) and the Low Priority Polling List (LPPL). The HPPL consists of high priority traffic from class 3 and class 4. The LPPL consists of low priority traffic from class 1 and class 2. In DWFSS the polling lists are sorted in descending order of TSID so that higher priority TS can get access to the medium earlier than low priority traffic. The HPPL is polled in the descending order as in the list. However, the LPPL is polled based on the MSI. If the MSI of the TS in the polling list will not expire during the next scheduled SI, then the TS is skipped and next TS in the LPPL is checked. A traffic stream will not be kept for more than one scheduled SI without access to the medium to prevent from starvation. Therefore, low priority traffic with MSI greater than two scheduled SI will be polled every other scheduled SI when the network load increases. However, when there is less high priority traffic, more low priority traffic is polled.

3.4 Admission Control Algorithm

In 802.11e admission control is performed based on a FCFS algorithm [8,24]. The FCFS algorithm is unable to provide fairness and prevent starvation. In order to overcome this problem faced by the 802.11e ACU, the DWFSS performs admission based on the dynamic weighted class-based bandwidth allocation model described earlier. The admission of newly requesting traffic flows are performed based on the admission control algorithm described in Fig. 5.

The main objective of admission control algorithm is to prevent starvation by preventing one type of traffic occupying the whole contention free bandwidth (HCCA bandwidth). To achieve this objective, the total contention free bandwidth is distributed among four classes. Each class is allocated a certain percentage of the total bandwidth. The ACU checks for the available bandwidth in the class before admitting a TS. If the available bandwidth in the class is not enough then the class is allowed to get some unused bandwidth from other classes. The other class offers a percentage of the unused bandwidth to the requesting class. An offering class does not offer all the bandwidth available in it to a requesting class. In order to admit more low priority traffic streams, the MSI of low priority traffic streams are considered. Since low priority traffic is not delay sensitive the DWFSS accepts twice more low priority traffic, if the low priority traffic can wait until another service interval. The extra low priority traffic will be polled during the next service interval. In determining the minimum and maximum acceptable values, it is important to highlight the extra low priority traffic admitted. The minimum and maximum acceptable values are determined, considering the extra low priority traffic. Two extreme cases are possible, and they are:

```

Algorithm:      ADMISSION_CONTROL
Input:         Class ID, scheduled SI
Output:        True / False

1. if unusedInClass greater than or equal to txop then
2.   reject ← false
3. else if high priority then
4.   if totalHP less than or equal to 75% and totalAvailable > txop then
5.     if weight_adjustment is true then
6.       reject ← false
7.     else
8.       reject ← true
9.     end if
10.  else
11.    reject ← true
12.  end if
13. else
14.  if totalLP less than or equal to 25% then
15.    weight_adjustment
16.    reject ← false
17.  else if totalHP greater than or equal to 52% and totalAvailable > txop then
18.    if weight_adjustment is true then
19.      reject ← false
20.    else if maxSI > scheduledSI x 2 then
21.      reject ← false
22.    else
23.      reject ← true
24.    end if
25.  end if
26.  else if totalLP less than or equal to 50% and maxSI > scheduledSI x 2 then
27.    reject ← false
28.    if totalLP less than or equal to 48% then
29.      weight_adjustment
30.    end if
31.  else if totalLP less than or equal to 96% and totalAvailable > txop then
32.    if maxSI > scheduledSI x 2 then
33.      reject ← false
34.    else
35.      reject ← true
36.    end if
37.  else
38.    reject ← true
39.  end if
40. return reject

```

Fig. 5 – The Admission Control Algorithm

Case 1: The maximum allowed high priority traffic of 75% has been admitted and low priority traffic is requesting for 50%. As in Table 2 DWFSS can still admit the minimum guaranteed 50% of low priority traffic. The actual allocation of HCCA bandwidth in this case is 75% and 25% for high priority and low priority traffic respectively in every service interval. The remaining 25% of low priority traffic will be scheduled in the next service interval (25% is served in every other service interval).

Case 2: The maximum allowed low priority 96% has been admitted and high priority is requesting for the minimum guaranteed 52%. In this case the actual allocation of service interval bandwidth will be 52% and 48% for high priority and low priority traffic respectively in every service interval. The remaining 48% of low priority traffic will be scheduled in the next service interval (48% is served in every other service interval).

In DWFSS high priority traffic is admitted when the class bandwidth is sufficient or if it can get sufficient bandwidth from other classes as described earlier. However, if a low priority traffic flow can wait for more than one service interval, then, the flow is admitted, even if other classes did not offer enough bandwidth. In other words, if DWFSS can maintain the delay requirements of a low priority TS, then the TS is accepted.

4.0 SIMULATION RESULTS AND DISCUSSIONS

In this section we present the simulations conducted to evaluate the proposed DWFSS. The simulations are conducted in NCTUns network simulator. Section 5.1 describes the simulation plan, Section 5.2 describes the traffic model, Section 5.3 describes the simulation parameters, Section 5.4 describes the performance metrics and Section 5.5 demonstrates the simulation results.

4.1 Simulation Plan and Environment

The simulation environment is developed using NCTUns network simulator. The evaluations are conducted on infrastructure wireless network topologies which consist of a QAP and multiple wireless QSTAs. The number of QSTAs and wired nodes vary between 1 and 15. Each QSTA run a traffic application which resembles VoIP, video, HTTP or bulk data in order to model a converged network of multiple types of traffic. The performance differences are compared based on the 802.11e with 802.11b PHY layer provided by NCTUns.

4.2 Traffic Model

Four different traffic applications which model VoIP, video, web and bulk data traffic representing different priority classes have been used. The VoIP and video traffic are considered as high priority traffic, HTTP and bulk data traffic as low priority traffic. For both the VoIP and video applications two way traffic is generated, to model voice conversations and video conferencing respectively. The VoIP and video TSPEC are modeled as specified by Harju *et al.* [29]. The Real-Time Transport Protocol (RTP) [30] over IP/UDP is used to carry the VoIP and video packets. The HTTP and bulk data traffic is modeled based on the parameters used by Zhang & Michelson [31], Yang [32] and Touch, Heidemann & Obraczka [33]. The G.729 codec was used to generate VoIP traffic and H.263 codec was used to generate video traffic. The TSPEC of each TS is specified in Table 3.

Table 3 – Traffic Specifications used

Traffic Type	Mean Data Rate (KB/s)	MSDU Size (Bytes)	MSI (ms)
VoIP	3	60	30
Video	56	1563	40
HTTP	25	150	60
Bulk data	25	1024	80

4.3 Simulation Parameters

In order to evaluate the performance of DWFSS mechanism and SS, the number of TS was increased linearly with the number of QSTAs increased. The Table 4 shows the simulation parameters used.

Table 4 – Simulation Parameters

Parameter	Value
Physical Layer	IEEE 802.11b
PHY rate	11 Mbps
Beacon Interval	100 ms
DIFS	50 μ s
PIFS	30 μ s
SIFS	10 μ s
Slot Time	20 μ s
CFP Max Duration	30 ms
TXOP Limit	8160 μ s

4.4 Performance Evaluation Metrics

The performance of DWFSS is evaluated based on the four metrics specified below:

1. Throughput: The number of bits correctly transferred from source to destination. The throughput is represented in Kbps.
2. End-to-end delay: The time taken between inserting a packet in the sender's queue and the reception of this packet by the receiver. The end-to-end delay is represented in milliseconds (ms).
3. Jitter: The difference of the latency of the current packet and previous packet. The jitter is represented in milliseconds (ms).
4. Fairness: Fairness is measured as the number of admitted flows compared to the number of requests made. The objective of this metric is to identify the fairness of admission control unit among different type of traffic.

4.5 Simulation Results

This section presents the simulation results that compare the existing IEEE 802.11e and the proposed DWFSS mechanism. The performance of both schedulers is compared in terms of throughput, end-to-end delay, jitter variation and fairness based on the defined metrics earlier. The number of flows and QSTAs are increased, in order to observe the effects of increasing number of QSTAs on each metrics. Table 5 shows the number of flows and type of traffic used in the simulation.

Table 5 – Number of flows used in Simulation

Flows	VoIP	Video	HTTP	Bulk data
4	1	1	1	1
6	1	2	2	1
8	2	3	2	1
10	3	3	2	2

All the simulations are conducted for a period of 60 seconds. A maximum of 10 flows is used in simulations, because the SS starts to reject any newly requesting flows more than 10, due to the size of CFP length and admitted TXOPs values. However, DWFSS can accept more low priority traffic flow.

4.5.1 Throughput

In the first set of simulation sessions the throughput of all the TS are measured. The objective of this experiment is to study the effect of increasing the number of flows to throughput. Fig. 6 shows the measured average one way throughput of DWFSS, SS and EDCA with increasing network load. The network starts to get congested when there are more than 8 QSTAs transmitting. The throughput of DWFSS and SS gradually increase till the network is congested. The throughput of TCP flow starts to decrease when the network is congested. The overall throughput decreased due the changes in TCP throughput. There were no significant changes to UDP throughput. However, the performance of TCP traffic is slightly better with DWFSS scheme. This improvement is visible on the graph in Fig. 6. The average throughput of DWFSS is 13.48% higher when compared to SS scheme. It is also noticeable that the throughput of DWFSS is higher when there are fewer flows. This increase in throughput is due to the increase in number of polls to the low priority traffic. In Fig. 6 a significantly higher throughput is observed under EDCA mechanism. The higher throughput of EDCA is due to two reasons. Firstly, the inefficient polling mechanism increases the overhead. The HCCA polling mechanism is inefficient due to the extra frames such as CF-Poll frames and QoS Null frames [34,35,36]. Secondly, due to the nature of TCP traffic applications in NCTUns. The more bandwidth offered, the more greedy TCP applications become. However, the performance of TCP applications under EDCA starts to decrease when the network load increases due to increase in collisions. Thus, the overall throughput under EDCA plummets when the network load increases as shown in Fig. 6.

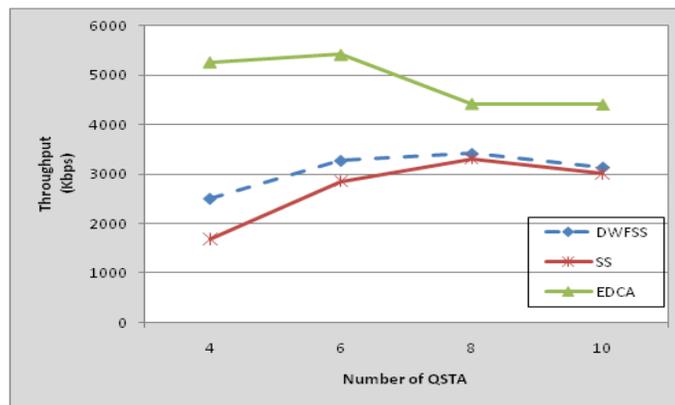


Fig. 6 – Throughput vs. number of QSTAs

4.5.2 End-to-end delay

In the second set of simulation sessions the end-to-end delay of VoIP and video streams are measured. The objective of this experiment is to study the delay performance of real-time multimedia traffic in the presence of other traffic. Fig. 7 shows the measured average end-to-end delay experienced by multimedia traffic for DWFSS, SS and EDCA mechanism. The end-to-end delay for EDCA significantly higher compared to DWFSS and SS. This higher delay is because of the contention base channel access mechanism and backoff procedure. The VoIP and video flows needs to compete with HTTP and bulk data traffic to gain access to the medium. Additionally, if another QSTA with low priority traffic is transmitting then the QSTA with VoIP or video traffic is required to backoff. If the waiting time is high compared to the data rate of application, then the delay increases. Thus, in EDCA the VoIP and video packets have to wait in the queue for a period of 20 milliseconds or more.

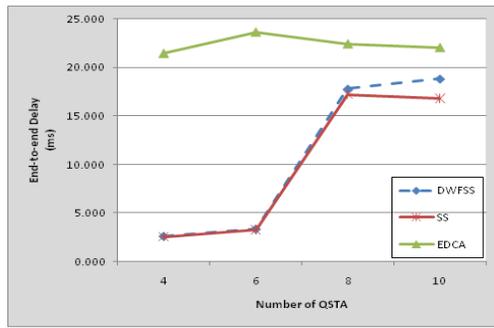


Fig. 7 – End-to-end delay vs. number of QSTAs

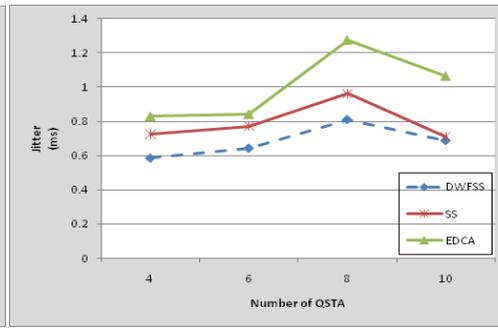


Fig. 8 – Jitter vs. number of QSTAs

4.5.3 Jitter

In the third set of simulation sessions the jitter experienced by VoIP and video streams are measured. The objective of this experiment is to study the jitter performance of multimedia traffic in the presence of other traffic. Fig. 8 shows the measured average jitter experienced by multimedia traffic for DWFFS, SS and EDCA mechanism. To maintain the QoE, the maximum acceptable jitter is 1 milliseconds [28]. Fig. 8 shows that the jitter of DWFFS is significantly lower than SS and EDCA in all cases. The average jitter of DWFFS is 16.21% lower when compared to SS scheme. The SS scheme performed better than EDCA by maintaining jitter below the maximum acceptable value 1 millisecond. However, EDCA mechanism was unable to maintain an acceptable level of jitter when the network load increased. It is also noticeable that when the network is congested with 10 QSTAs, the jitter for all the three schemes decreased. This decrease is caused by the TCP congestion control mechanism. As the TCP flows decrease their transmission rate, the multimedia traffic streams gets access to the medium more often. The rapid increase in jitter from 6 to 8 flows is due to the VBR video application used in the simulation. As VBR applications generate packets at a variable rate with different sizes, the TXOP calculated during admission may not be enough to transmit all the packets in the QSTA buffer [37].

4.5.4 Fairness

In measuring the fairness of admission control, only DWFFS and SS mechanism are compared. Two experiments were conducted, first one with low priority HTTP traffic starting before all the other traffic and the second one with high priority video flows starting before all the other traffic. The objective of the two experiments is to study the fairness of the algorithms in allocating bandwidth to different traffic when some traffic flows arrived earlier than the others.

4.5.4.1 Fairness Test 1

In the first experiment ten HTTP flows were started first, followed by two VoIP, two video and one bulk data flows. Table 6 shows the number of flows started and the number flows accepted by DWFFS and SS mechanism for this experiment. The results in Table 6 and Figure 9 shows that SS scheme accepted all the HTTP flows but rejected one VoIP, two video, and one bulk data flows. The video and bulk data flows starve under SS scheme due to the FCFS admission control mechanism. However, the DWFFS mechanism, accepted all the flows without rejecting any TS as shown in Fig. 10.

Table 6 – Number of flows admitted

	VoIP	Video	HTTP	Bulk Data	Total
Requested	2	2	10	1	15
Accepted by SS	1	0	10	0	11
Accepted by DWFFS	2	2	10	1	15

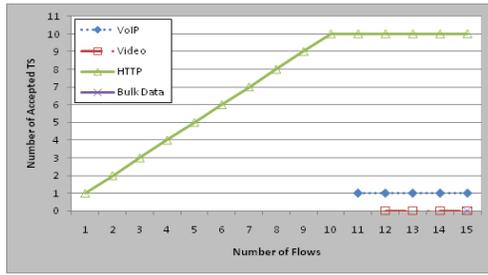


Fig. 9 – Number of flows admitted by SS

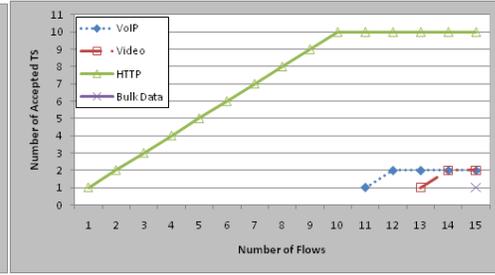


Fig. 10 – Number of flows admitted by DWFSS

4.5.4.2 Fairness Test 2

In the second experiment eight video flows were started first, followed by two VoIP, two HTTP and one bulk data flows. Table 7 shows the number of TS started and the number TS accepted by DWFSS and SS mechanism. The results in Table 7 and Fig. 11 show that SS scheme accepted seven flows, which is the maximum number of possible video flows and one VoIP flow. One video flow was rejected and one VoIP flow was admitted because the TXOP of video flow is larger than the VoIP flow. The SS scheme rejected all the other TS leading to starvation of HTTP and bulk data flows. Fig. 12 shows that DWFSS accepted only five video flows and did not let the video flows to occupy all the bandwidth. The DWFSS accepted two VoIP, two HTTP and one bulk data flow. As shown in Table 7 DWFSS rejected three video flows and accepted 10 flows. These results show that DWFSS can provide fairness by reserving bandwidth for different classes of traffic. Additionally, this result also shows that DWFSS can admit more flows compared to SS scheme.

Table 7 – Number of flows admitted

	VoIP	Video	HTTP	Bulk Data	Total
Requested	2	8	2	1	13
Accepted by SS	1	7	0	0	8
Accepted by DWFSS	2	5	2	1	10

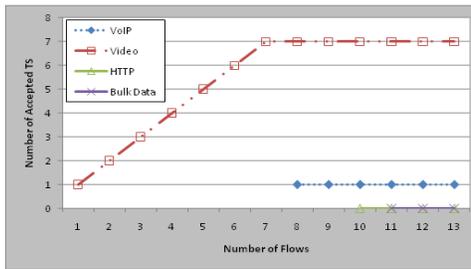


Fig. 11 – Number of flows admitted by SS

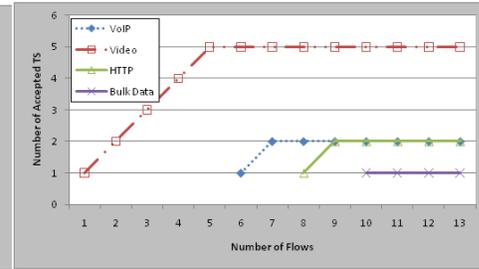


Fig. 12 – Number of flows admitted by DWFSS

5.0 CONCLUSION

In this paper we have proposed an admission control and scheduling algorithm called Dynamic Weighted Fair Scheduling Scheme (DWFSS) to enhance the IEEE 802.11e. Two priority lists, namely the High Priority Polling List and the Low Priority Polling, have been introduced to differentiate high priority traffic and low priority traffic in providing services. A new admission control algorithm which is able to admit more low priority traffic has been added. The proposed DWFSS mechanism successfully improves the fairness among different traffic classes and can improve QoE in a converged network. As the simulation results showed the QoS performance of DWFSS mechanism is better than the standard scheduler. Even though, the throughput of EDCA was higher in our simulations, delay and jitter performance of multimedia traffic was poor under EDCA. The DWFSS managed to improve throughput and jitter compared to the standard scheduler. The DWFSS maintained an acceptable level of delay which is slightly higher than the standard scheduler. However, when compared to standard scheduler, DWFSS outperforms in terms of fairness and prevented starvation.

6.0 REFERENCES

- [1] IEEE, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," New York 1999.
- [2] L. W. Lim, R. Malik, P. Y. Tan, C. Apichaichalermwongse, K. Ando, and Y. Harada, "A QoS scheduler for IEEE 802.11e WLANs," in Consumer Communications and Networking Conference, 2004. CCNC 2004. First IEEE, 2004, pp. 199-204.
- [3] A. Petcher. (2006). QoS in Wireless Data Networks. [Online]. Available: http://www.cs.wustl.edu/~jain/cse574-06/ftp/wireless_qos/index.html.
- [4] J. Wroclawski, "The Use of RSVP with IETF Integrated Services," RFC 2210, September 1997, 1997.
- [5] D. Grossman, "RFC3260: New Terminology and Clarifications for Diffserv," Internet RFCs, 2002.
- [6] IEEE, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications," IEEE, New York 2005.
- [7] R. Lenagala and Q. A. Zeng, "Starvation Prevention Scheme for the IEEE 802.11e EDCF Using Dynamic MAC Layer Parameters," in Wireless Telecommunications Symposium, 2006. WTS '06, 2006, pp. 1-7.
- [8] B. A. Venkatakrishnan and S. Selvakennedy, "An Enhanced HCF for IEEE 802.11e Wireless Networks," in Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems, Venice, Italy, 2004.
- [9] D. Gao, J. Cai, and L. Zhang, "Physical rate based admission control for HCCA in IEEE 802.11e WLANs," in Advanced Information Networking and Applications, 2005. AINA 2005. 19th International Conference on, 2005, vol. 1, pp. 479-483 vol.471.
- [10] F. Didi, M. Feham, H. Labiod, and G. Pujolle, "Dynamic Admission Control Algorithm for WLANs 802.11," in Information and Communication Technologies: From Theory to Applications, 2008. ICTTA 2008. 3rd International Conference on, 2008, pp. 1-6.
- [11] Z. A. B. M. Noh, T. Suzuk, and S. Tasaka, "A Multimedia Priority Dynamic Scheduling Scheme for Audio-Video Transmission with User-Level QoS Guarantee by IEEE 802.11e HCCA," in Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on, 2007, pp. 1-5.
- [12] Z. A. Bin Muhamad Noh, T. Suzuk, and S. Tasaka, "Guaranteeing QoE in Audio-Video Transmission by IEEE 802.11 e HCCA," IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences, vol. 91, pp. 1551-1561, 2008.
- [13] D. Skyrianoglou, N. Passas, and A. Salkintzis, "Traffic scheduling for multimedia QoS over wireless LANs," in Communications, 2005. ICC 2005. 2005 IEEE International Conference on, 2005, vol. 2, pp. 1266-1270 Vol. 1262.
- [14] Y.-C. Chen and H.-R. Yeh, "An Adaptive Polling Scheme Supporting Audio/Video Streaming in Wireless LANs," in Future Trends of Distributed Computing Systems, 2008. FTDCS '08. 12th IEEE International Workshop on, 2008, pp. 16-22.
- [15] T. D. Lagkas, G. I. Papadimitriou, P. Nicopolitidis, and A. S. Pomportsis, "Priority-Oriented Adaptive Control With QoS Guarantee for Wireless LANs," Vehicular Technology, IEEE Transactions on, vol. 56, pp. 1761-1772, 2007.
- [16] D. Gozalvez, J. F. Monserrat, J. Gozalvez, and L. Rubio, "An Efficient HCF Scheduling Mechanism in Mixed Traffic Scenarios," in Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th, 2007b, pp. 1026-1030.
- [17] W.-K. Kuo, "Traffic scheduling for multimedia transmission over IEEE 802.11e wireless LAN," Communications, IET, vol. 2, pp. 92-97, 2008.
- [18] S. M. Kim and Y. J. Cho, "Channel time allocation scheme based on feedback information in IEEE 802.11 e wireless LANs," Computer Networks, vol. 51, pp. 2771-2787, 2007.
- [19] S. Shi, G. Zhu, and G. Su, "An Optimized QoS Traffic-Scheduling Algorithm based on HCCA," in Intelligent Computation Technology and Automation (ICICTA), 2008 International Conference on, 2008, vol. 2, pp. 229-233.
- [20] A. R. Elshafei and U. Baroudi, "Wireless Fair Queuing algorithm for window-based Link Level Retransmission," in Computer Systems and Applications, 2008. AICCSA 2008. IEEE/ACS International Conference on, 2008, pp. 386-391.
- [21] N. Kim and H. Yoon, "Wireless packet fair queueing algorithms with link level retransmission," Computer Communications, vol. 28, pp. 713-725, 2005.
- [22] V. A. Siris and G. Stamatakis, "A dynamic CBWFQ scheme for service differentiation in WLANs," in Personal, Indoor and Mobile Radio Communications, 2005. PIMRC 2005. IEEE 16th International Symposium on, 2005, vol. 4, pp. 2665-2669 Vol. 2664.

- [23] G. Boggia, P. Camarda, L. A. Grieco, and S. Mascolo, "Feedback-based bandwidth allocation with call admission control for providing delay guarantees in IEEE 802.11 e networks," *Computer Communications*, vol. 28, pp. 325-337, 2005.
- [24] I. A. Qaimkhani and E. Hossain, "Efficient silence suppression and call admission control through contention-free medium access for VoIP in WiFi networks," *Communications Magazine, IEEE*, vol. 46, pp. 90-99, 2008.
- [25] A. Demers, S. Keshav, and S. Shenker, "Analysis and simulation of a fair queueing algorithm," *Applications, Technologies, Architectures, and Protocols for Computer Communication*, pp. 1-12, 1989.
- [26] P. Kun, L. Xiaokang, Z. Junli, and G. Xuedao, "Dynamic WFQ scheduling for real-time traffic in wireless ATM links," in *Communication Technology Proceedings, 2000. WCC - ICCT 2000. International Conference on, 2000*, vol. 2, pp. 1479-1482 vol.1472.
- [27] Cisco. (2005). *Enterprise QoS Solution Reference Network Design Guide*. [Online]. Available: <http://www.cisco.com/univercd/cc/td/doc/solution/esm/qosrnd.pdf>.
- [28] ITU, "End-user multimedia QoS categories - ITU-T Recommendation G.1010," *International Telecommunication Union 2001*.
- [29] J. Harju, G. Heijenk, P. Langendörfer, and V. A. S. (Eds.), "Wired/Wireless Internet Communications," in *6th International Conference, WWIC 2008, Proceedings, Tampere, Finland, 2008*, vol. 5031.
- [30] H. Schulzrinne, G. Fokus, and S. Casner, "RTP: A Transport Protocol for Real-Time Applications," *IETF, R. F. C.* 1889, 1996.
- [31] Y. Zhang and D. G. Michelson, "Multi-Service Opportunistic QoS-Enhanced Scheduler for the Downlink of IEEE 802.16 Point-to," *WiMax/MobileFi: Advanced Research And Technology, 2007*.
- [32] L. T. Yang, *Embedded Software and Systems: Second International Conference, ICES 2005, Xi'an, China, December 16-18, 2005: Proceedings: Springer, 2005*.
- [33] J. Touch, J. Heidemann, and K. Obraczka, "Analysis of HTTP Performance," URL: <http://www.isi.edu/touch/pubs/http-perf96/>, ISI Research Report ISI/RR-98-463,(original report dated Aug. 1996), USC/Information Sciences Institute, 1998.
- [34] W. Ping, J. Hai, and Z. Weihua, "IEEE 802.11e enhancement for voice service," *Wireless Communications, IEEE*, vol. 13, pp. 30-35, 2006.
- [35] L. Il-Gu, S. Jung-Bo, Y. Sung-Rok, and P. Sin-Chong, "Efficient block size based polling scheme for IEEE 802.11e wireless LANs," in *Vehicular Technology Conference, 2005. VTC 2005-Spring. 2005 IEEE 61st, 2005*, vol. 5, pp. 2869-2873 Vol. 2865.
- [36] D. Gozalvez, J. F. Monserrat, D. Calabuig, and J. Gozalvez, "Policy-based channel access mechanism selection for QoS provision in IEEE 802.11e," *Vehicular Technology Magazine, IEEE*, vol. 2, pp. 29-34, 2007a.
- [37] M. M. Rashid, E. Hossain, and V. K. Bhargava, "Queueing Analysis of 802.11e HCCA with Variable Bit Rate Traffic," in *Communications, 2006. ICC '06. IEEE International Conference on, 2006*, vol. 10, pp. 4792-4798.

BIOGRAPHY

TC Ling is an Associate Professor at the Faculty of Computer Science & Information Technology, University of Malaya, Malaysia. He obtained his PhD from the University of Malaya. He teaches computer networks. His current research interests include high speed computer networks, cloud computing, mobile ad hoc networks, inter domain network QoS and wireless security.

Ahmed Riza obtained his Master of Computer Science from the University of Malaya in 2009. He is currently a networker.

KK Phang is an Associate Professor at the Faculty of Computer Science & Information Technology, University of Malaya, Malaysia. He obtained his PhD from the University of Malaya in 2004. He teaches computer networks and distributed systems. His current research interests include high speed computer networks, network quality of service, mobile ad hoc networks and distributed systems.

Hoh Kian Pee obtained his B. Engg (Elec) in 1986 from University of Malaya and MBA from Herriot Watt University, UK in 1994. Currently, he is a Lecturer at the Information Technology Department, School of Electronics and Information Technology at Institute of Technology, College East, Singapore