Provisioning of Parameterized Quality of Service in 802.11e Based Wireless Mesh Networks

Xiaowen Chu
Department of Computer Science, Hong Kong Baptist University
Kowloon, Hong Kong, P. R. China
Email: chxw@comp.hkbu.edu.hk

Abstract

There has been a growing interest in the use of wireless mesh networks. Today's wireless technology enables very high data rate up to hundreds of Megabits per second, which creates the high demand of supporting real-time multimedia applications over wireless mesh networks. Hence it is imperative to support quality of service (QoS) in wireless mesh networks. In this paper, we design a framework to provide parameterized QoS in 802.11e based wireless mesh networks. Our framework consists of admission control algorithms and scheduling algorithms, which aim at supporting constant bit-rate (CBR) traffic flows, as well as variable bit-rate (VBR) traffic flows. We first present deterministic end-to-end delay bounds for CBR traffic. We then prove that the delay of VBR traffic can be bounded if the traffic flow conforms to a leaky-bucket regulator. We further study different admission control algorithms for VBR traffic. Our simulation results show that, by taking advantage of statistical multiplexing, much more traffic flows can be admitted.

I. Introduction

Wireless mesh networks (WMNs) are gaining significant progress in both academia research and commercial deployment in recent years [1, 2], which find numerous applications such as community networks, broadband home networking, enterprise networking, etc. In [3], the development of a wireless community network based on IEEE 802.11b was presented. In

[4], Karrer et al. proposed a new wireless backbone infrastructure based on wire-powered Transit Access Points (TAPs) which deploy beamforming antennas. The measurement of 802.11b based urban wireless mesh network named Roofnet was presented in [5, 6]. A testbed with 23 802.11a nodes was constructed for the purpose of evaluating different routing metrics [7, 8], with an important feature that multi-radio technique is used to boost the network throughput. End-to-end performance issue in wireless mesh networks, especially the TCP fairness problem, was investigated in [9]. The measurement study of an infrastructure WMN backbone was reported in [10]. The feasibility of WMN was studied in [11] based on a deployed system and trace-based evaluation. Recently, it has been shown that the aggregated system throughput can be significantly improved by exploiting multi-channel and multi-interface technique [12, 13, 14, 15, 16, 17, 18].

This paper primarily focuses on infrastructure WMNs based on IEEE 802.11 technologies [19], which provide a wireless backbone network to connect numerous wireless clients in a large area. A typical example of infrastructure WMN is shown in Fig. 1, whose backbone consists of a set of wireless mesh routers. Wireless stations (i.e., end users) can access the Internet by associating with a nearby wireless mesh router. Given that IEEE 802.11b/g provides 3 non-overlapping channels and 802.11a provides 12 non-overlapping channels [20, 21, 22], we make the following assumption in the WMNs studied in this paper: the interfaces of a wireless mesh router use non-overlapping channels so that all the interfaces can operate simultaneously.

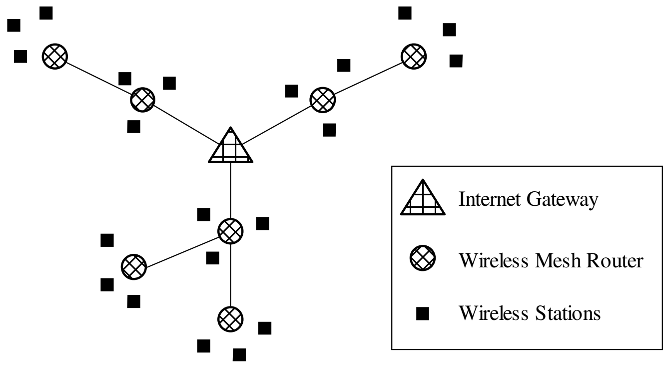

Fig. 1: An example of infrastructure wireless mesh network

Existing research on WMNs has been mainly focused on the improvement of system throughput. Although QoS provisioning has been extensively studied for traditional wireless networks such as WLAN [23], little work has been done to address the QoS issue in WMNs. There are generally two different QoS models, namely prioritized QoS and parameterized QoS. In prioritized QoS model, the QoS requirements are specified as a set of priorities, and the network treats the traffic based on the priority associated with each packet; in parameterized QoS model, QoS requirements are specified as a set of parameters, such as bandwidth, delay, delay jitter, packet loss rate, etc., and the network treats the traffic flows based on the set of QoS parameters negotiated by the flow initiator and the network. Recently, a prioritized QoS framework named Wireless DiffServ has been proposed in [24], which exploits the feature of IEEE 802.11e EDCA [25] to provide differentiated QoS services. While providing parameterized QoS in traditional wireless ad hoc networks has been shown to be extremely difficult due to the varying network topology, channel capacity, mobility of wireless stations, etc. [26], it is possible to provide parameterized QoS in infrastructure WMNs because of the stable wireless backbone. This paper complements the Wireless DiffServ by designing a framework for the provisioning of parameterized QoS in WMNs based on the IEEE 802.11e HCCA protocol. Different from the Wireless DiffServ, call admission control and resource reservation are the necessary components in order to provide parameterized QoS.

The rest of the paper is organized as follows. We first review the IEEE 802.11e QoS mechanisms in Section II. Section III presents the framework of provisioning parameterized QoS for both CBR flows and VBR flows. Section IV and V present the mechanisms of QoS provisioning for CBR flows and VBR flows, respectively. Section VI presents the simulation results. Finally, Section VII concludes the paper with our future research plan.

II. Review of IEEE 802.11e QoS Mechanisms

A new MAC layer function named Hybrid Coordination Function (HCF) is proposed in 802.11e, which multiplexes between two medium access modes: a distributed contention-based scheme named Enhanced distributed Channel Access (EDCA), and a centralized polling-based scheme named HCF Controlled Channel Access (HCCA). The Access Point (AP) and wireless stations (STAs) that implement the QoS facilities are called QoS-enhanced AP (QAP) and QoS-enhanced STAs (QSTAs), respectively.

Under HCF, the basic unit of allocation of the right to transmit onto the wireless medium is the transmission opportunity (TXOP). Each TXOP is defined by a starting time and a defined maximum length. The TXOP may be obtained by a QSTA winning an instance of EDCA contention during the contention period (CP), or by a non-AP QSTA receiving a poll frame. The former is called EDCA TXOP, while the latter is called HCCA TXOP. A QSTA is allowed to transmit a burst of data frames within its TXOP allocation.

A. Contention-based Channel Access: 802.11e EDCA

In EDCA, the QoS support is achieved by introducing four access categories (ACs) in each QSTA. Different ACs have different priorities. Each data packet from the higher layer along with a specific user priority value should be mapped into a corresponding AC. Each AC has its own transmission queue, and behaves as a single DCF contending entity with its own access parameters. The differentiation in priority is achieved by setting different values for a set of access parameters, including arbitrary interframe space number, contention window, and TXOP limit. The AC with the smallest backoff wins the internal contention and uses the backoff value to contend externally for the wireless medium. If the backoff counters of two or more ACs in one STA reach zero simultaneously, a virtual collision occurs, and the TXOP is given to the AC with the highest priority among the colliding ACs, and the other colliding ACs defer and try again later.

B. Contention-free Channel Access: 802.11e HCCA

802.11e HCCA is a centralized polling scheme that aims at providing parameterized QoS support. The concept of traffic streams (TSs) is introduced in HCCA. Before any data transmission, a TS is first established, and each QSTA can have no more than eight TSs with different priorities. In order to initiate a TS connection, a QSTA sends a QoS request frame containing a traffic specification (TSPEC) to the QAP. A TSPEC describes the QoS requirements of a TS, such as mean/peak data rate, mean/maximum frame size, delay bound, and a maximum required service interval (RSI). A maximum RSI refers to the maximum duration between the start of successive TXOPs that can be tolerated by a requesting application. On receiving all these QoS requests, the QAP scheduler first determines the selected service interval (SI), which should be the highest submultiple value of the beacon interval and also be no larger than all the maximum RSIs required by the different TSs from different QSTAs. Then an 802.11e beacon interval is divided into an integer number of SIs, and QSTAs are polled sequentially during each selected SI, as illustrated in Fig. 2. In this way, all the admitted TSs should be polled once within the delay requirement of the most time-stringent TS. Lastly, the QAP scheduler computes the corresponding HCCA-TXOP values for different QSTAs by using their QoS requests in TSPECs, and allocates them to those QSTAs.

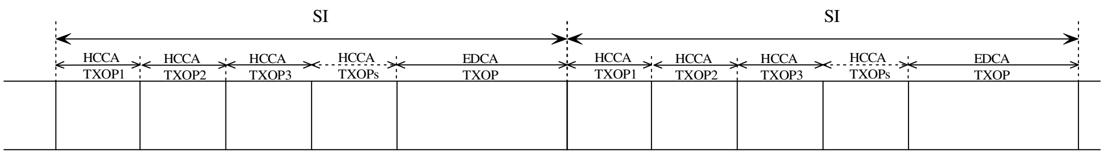

Fig. 2: HCCA Channel Access Mechanism

The QAP polls QSTAs for a TXOP duration, which is calculated from reservation requests sent by the QSTA. The TXOP is initiated by a poll request from the QAP and during this duration, transmissions can occur in both the uplink and downlink directions. The TXOP ends if one of the following conditions occurs: neither the QAP nor the QSTA have any frames left to transmit, the channel idle time has exceeded the timeout period, or the TXOP duration has expired.

An HCCA admission control algorithm is suggested in 802.11e. Using the TSPEC information, the QAP calculates a ratio of the transmission time reserved for HCCA of all existing K QSTAs over an SI: $\sum_{i=1}^{K} \frac{TXOP_i}{T_{SI}}$. In order to decide whether or not a request from a new traffic flow can be accepted in HCCA, the QAP scheduler only needs to check if the new request $TXOP_{K+1}$ plus all the current TXOP allocations are lower than or equal to the maximum fraction of time that can be used by HCCA: $\frac{TXOP_{K+1}}{T_{SI}} + \sum_{i=1}^{K} \frac{TXOP_i}{T_{SI}} \le \frac{T_{CAPLimit}}{T_{Beacon}}$, where $T_{CAPLimit}$ is the maximum duration bound of HCCA, and T_{Beacon} represents the length of a beacon interval. More discussions about admission control in IEEE 802.11e can be found in [27, 28].

III. System Framework of QoS Provisioning

We assume all the wireless links in the wireless mesh network support IEEE 802.11e so that end-to-end parameterized QoS can be achieved. In order to setup a flow with guaranteed service, the wireless station first sends a request message to the access router through EDCA which specifies the traffic type and traffic specification. The access router examines the traffic specification and decides whether it can support this flow or not. If it can, the access router forwards the request message to the next router along the path towards the destination. The admission control process checks whether the new flow's QoS requirement can be met without affecting the QoS of exiting flows. More details of admission control will be elaborated in Section IV and Section V. It is commonly possible to use QoS routing algorithms to decrease the call blocking probability [29]. In this paper, we only consider the simple shortest path routing algorithm, and leave QoS routing as our future work. To shorten the admission control process, a higher EDCA priority can be assigned to the signaling messages.

In this framework, we distinguish the access wireless link from the backbone wireless link. For the access wireless link, the wireless router functions as the QAP. For a backbone wireless link, the wireless router which is closer to the Internet gateway functions as the QAP. As mentioned previously in Section I, we assume that any two neighboring wireless links use different channels so that a wireless router can have simultaneous data communication on all its interfaces.

The bandwidth of an access link is shared by a number of wireless stations and a wireless router, as shown in Fig. 3. A beacon is divided into several service intervals (SIs), and we assume all the links use the same value of T_{SI}. Within an SI, the QAP first grants HCCA TXOPs to those stations that have reserved bandwidth. The HCCA TXOP assigned to a station can include two parts: CTXOP for CBR flows, and VTXOP for VBR flows. After the HCCA period, the EDCA TXOPs are available for all the stations to send/receive best-effort data traffic. Prioritized QoS can be achieved by exploiting the feature provided by EDCA, but it is beyond the topic of this paper.

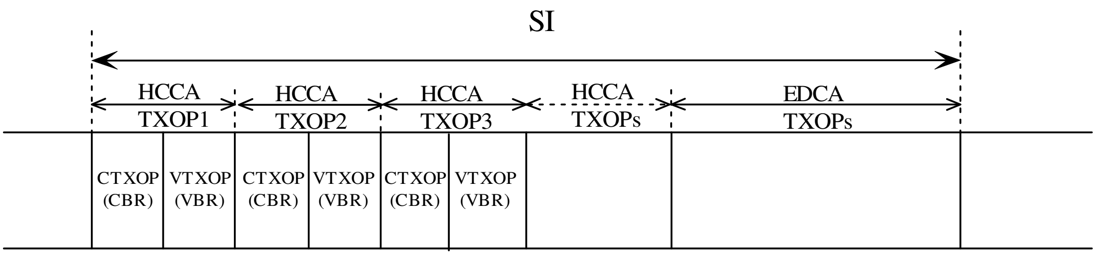

Fig. 3: HCCA Channel Access Mechanism at the access link

The bandwidth of a backbone wireless link is shared by a small number of wireless routers. By using Direct Link Protocol (DLP), a wireless router can send traffic to its neighbor directly without going through the HC who is in charge of this channel [24]. In Fig. 4, we show the bandwidth assignment on a backbone wireless link which is shared by only two routers. During TXOP1, the downlink router gets the chance to send packets to the uplink router; during TXOP2, the uplink router gets the chance to send packets to the downlink router. For each backbone link, the wireless router maintains a set of queues, each for a traffic flow, as shown in Fig. 5. Different scheduling algorithms can be designed for packet delivering.

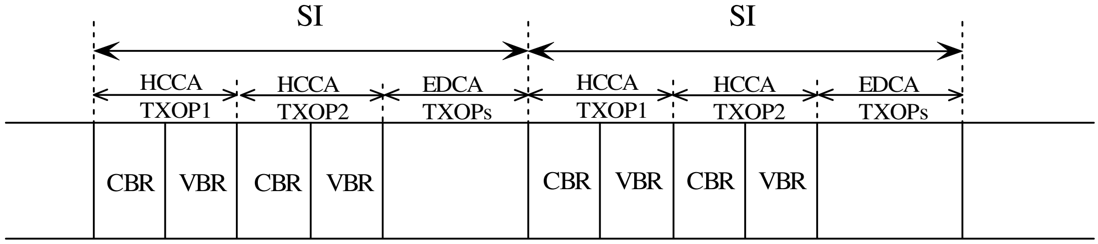

Fig. 4: HCCA channel access mechanism at a backbone link

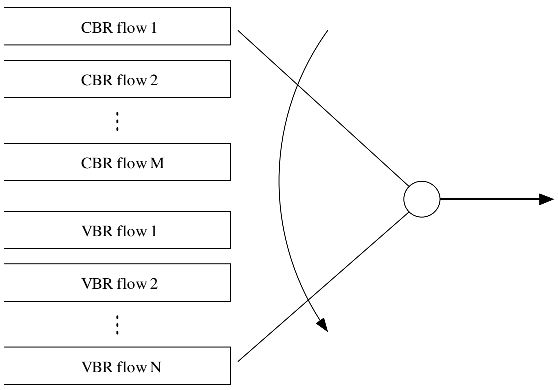

Fig. 5: Packet scheduling at a backbone link

Considering the time period of CBR traffic, VBR traffic, and best-effort traffic in each SI, we propose two different link capacity partition approaches: fixed TXOP assignment, and dynamic TXOP assignment.

By using fixed TXOP assignment, for each SI, a constant time period with length $R_{CBR} \cdot T_{SI}$ is assigned to CBR traffic; a constant time period with length $R_{VBR} \cdot T_{SI}$ is assigned to VBR traffic; and the remaining constant time period with length $(1 - R_{CBR} - R_{VBR}) \cdot T_{SI}$ is assigned to best-effort traffic. The corresponding scheduling algorithm is named fixed TXOP scheduling, which will be further discussed in Section V.A.

By using dynamic TXOP assignment, for each SI, a constant time period with length $R_{CBR} \cdot T_{SI}$ is assigned to CBR traffic; a variant time period with an average length of $R_{VBR\_avg} \cdot T_{SI}$ and maximum length of $R_{VBR\_max} \cdot T_{SI}$, is assigned to VBR traffic, under the condition that $0 \le R_{VBR\_avg} < R_{VBR\_max} < 1 - R_{CBR}$; and the remaining time period is used for EDCA. The rational of using a variant time period for VBR traffic is to make the system adaptive to the burstiness of VBR traffic: under heavy traffic, some bandwidth can be borrowed from EDCA and assigned to VBR traffic; while under light traffic, the borrowed bandwidth can be returned to EDCA traffic. The system keeps the average time period of VBR traffic as $R_{VBR\_avg} \cdot T_{SI}$ in a long term. The parameter R_{VBR_max} guarantees that at least $(1 - R_{CBR} - R_{VBR\_max}) \cdot T_{SI}$ of time are reserved for EDCA traffic in every SI, which can be used for signaling messages. The parameters R_{CBR}, R_{VBR_avg}, and R_{VBR_max} are configured by the network operators. The corresponding scheduling algorithm is named dynamic TXOP scheduling, which will be further discussed in Section V.B.

IV. QoS Support for CBR traffic

Assume the route between the source and destination consists of *k* hops, as shown in Fig. 6. The destination is the Internet Gateway, which is also the QAP of the last hop. Station *m* (node 0 in the path) requests a new CBR traffic flow $f_{m,j}$ with a constant data rate of $\rho_{m,j}$ and fixed packet size $l_{m,j}$. The minimum physical rate of the access link is C_0. The current CTXOP assigned to station *m* is $CTXOP_m'$. Router 1 is the access router, which gathers data from a number of attached source nodes. In the following, we present admission control algorithms for CBR traffic without and with delay requirement, respectively.

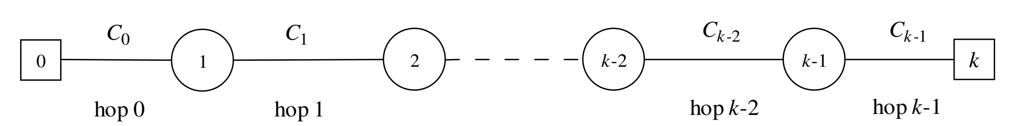

Fig. 6: a *k*-hop end-to-end path

A. Admission control for CBR traffic without delay requirement

If the CBR traffic $f_{m,j}$ has no delay requirement, the admission control algorithm simply checks the available bandwidth on all the links along the path:

1. The access router calculates the number of frames that arrive from flow $f_{m,j}$ during an SI: $N_{m,j} = \left\lceil \frac{\rho_{m,j} \cdot T_{SI}}{l_{m,j}} \right\rceil$;
2. The access router recalculates the TXOP duration requested by station *m*: $CTXOP_m = CTXOP_m{}' + N_{m,j}(\frac{l_{m,j}}{C_0} + O_0)$ where O_0 is the per-frame overhead on the access link, which includes interframe space, ACK frame, MAC and PHY overhead.
3. If $\sum_{i=1}^{K} CTXOP_i \le R_{CBR} \cdot T_{SI}$, goto Step 4; otherwise, reject the flow request.
4. The access router forwards the request message to the next router along the shortest path towards the destination.
5. Upon receiving a request message from router $i-1$, router i, $2 \le i \le k$, performs the following: Denote the original CTXOP assigned to router $i-1$ by $CTXOP_d$. If $CTXOP_d + N_{m,j}(\frac{l_{m,j}}{C_i} + O_i) + \sum_{s:\, s \ne d} CTXOP_s \le R_{CBR} \cdot T_{SI}$, where O_i is the per-frame overhead on link *i*, router *i* either forwards the request message to the next router ($i < k$), or returns an accept message to the previous router ($i = k$); otherwise, it returns a reject message to the source node through the intermediate routers.

B. Admission control for CBR traffic with delay requirement

In this section, we study admission control for CBR flows with delay requirement. Assume the flow requires a delay bound of *D*.

Still consider the route in Fig. 6. Assume the propagation delay at hop *i* is τ_i, where

$0 \le i \le k-1$; the processing delay at node *i* is ξ_i, where $1 \le i \le k$; and the queueing delay at node *i* is T_q^i, where $0 \le i \le k-1$.

The end-to-end delay, denoted by *T*, consists of the propagation delay T_{prop}, processing delay T_{proc}, transmission delay T_{tran}, and queueing delay T_q, where $T_{prop} = \sum_{i=0}^{k-1} \tau_i$, $T_{proc} = \sum_{i=1}^{k} \zeta_i$, $T_{tran} = \sum_{i=0}^{k-1} \frac{L}{C_i}$, $T_q = \sum_{i=0}^{k-1} T_q^i$.

At the source node, the CTXOP assigned to this flow is $N(\frac{L}{C_0} + O_0)$, where $N = \left\lceil \frac{\rho \cdot T_{SI}}{L} \right\rceil$. The worst case which results in the longest queueing delay at the source node is shown in Fig. 7: During the first SI, the QAP polls the source node at the beginning, but unfortunately the source node has no data to transmit at that time and hence the CTXOP in the first service interval is wasted. The elapsed time caused by the QoS CF-POLL frame and QoS NULL frame is denoted by κ_0. During the second SI, the QAP polls the node at the end of the $R_{CBR} \cdot SI$. Hence the largest delay is:

$(1+R_{CBR})T_{SI} - \kappa_0 - TXOP = (1+R_{CBR})T_{SI} - \kappa_0 - N(\frac{L}{C_0} + O_0) = (1+R_{CBR})T_{SI} - \frac{N \cdot L}{C_0} - O_0'$, where $N = \left\lceil \frac{\rho \cdot T_{SI}}{L} \right\rceil$, and $O_0' = \kappa_0 + N \cdot O_0$. Finally we have:

$$T_q^0 \le (1+R_{CBR})T_{SI} - \frac{N \cdot L}{C_0} - O_0'. \tag{1}$$

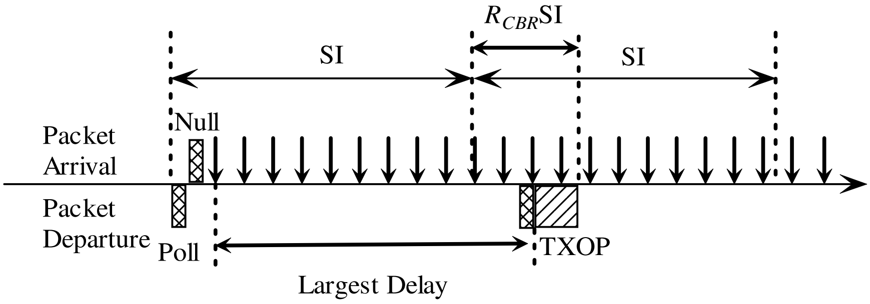

Fig. 7: The scenario of largest delay at hop 0

Next, we consider the i^{th} intermediate wireless mesh router, $1 \le i \le k-1$. At the incoming link (i.e., link $i-1$), the CTXOP assigned to this flow is $N \cdot (L/C_{i-1} + O_{i-1})$; at the outgoing link (i.e., link i), the CTXOP assigned to this flow is $N \cdot (L/C_i + O_i)$. The worst case that results in the longest queueing delay at this router is illustrated in Fig. 8: above the time axis, it shows the packet arrivals from the incoming link; and below the time axis, it shows the packet departures at the outgoing link. The largest packet delay equals $(1+R_{CBR})T_{SI} - \kappa_i - TXOP = (1+R_{CBR})T_{SI} - \frac{N \cdot L}{C_i} - O_i'$ where $O_i' = \kappa_i + N \cdot O_i$. Therefore we have:

$$T_q^i \le (1+R_{CBR})T_{SI} - \frac{N \cdot L}{C_i} - O_i' \quad \text{where} \quad 1 \le i \le k-1. \tag{2}$$

Finally, we have

$$\begin{aligned} T &= T_{prop} + T_{proc} + T_{tran} + T_q \\ &= \sum_{i=0}^{k-1} \tau_i + \sum_{i=1}^{k} \zeta_i + \sum_{i=0}^{k-1} \frac{L}{C_i} + \sum_{i=0}^{k-1} T_q^i \\ &\le \sum_{i=0}^{k-1} (\tau_i + \zeta_{i+1} + \frac{L}{C_i}) + (1+R_{CBR})T_{SI} - \frac{N \cdot L}{C_0} - O_0' + \sum_{i=1}^{k-1} ((1+R_{CBR})T_{SI} - \frac{N \cdot L}{C_i} - O_i') \\ &= \sum_{i=0}^{k-1} (\tau_i + \zeta_{i+1} + \frac{L}{C_i} - O_i') + k(1+R_{CBR})T_{SI} - N \cdot L \cdot \sum_{i=0}^{k-1} \frac{1}{C_i} \end{aligned} \tag{3}$$

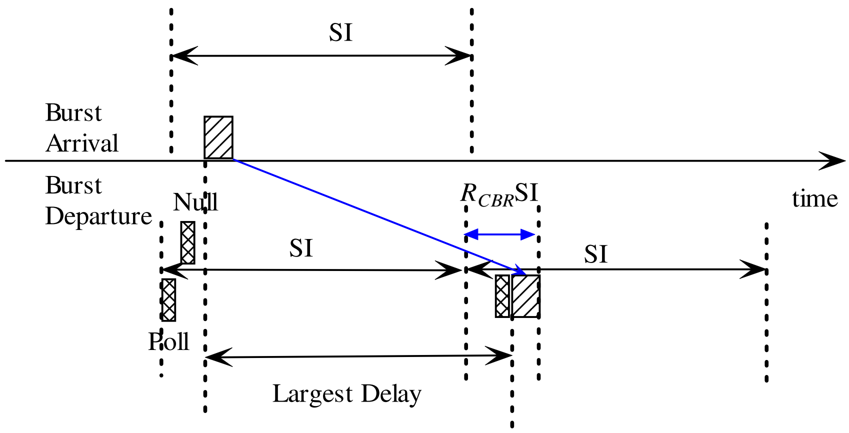

Fig. 8: The scenario of largest delay at an intermediate link

After deriving the end-to-end delay bound, we propose the admission control algorithm as follows:

1. Check the bandwidth availability using the admission control principle for CBR traffic without delay requirement.
2. Perform the following algorithm to check the delay requirement:

---

calculate T using Eq. 3.

set $SI_{\min} = \inf\{S : S \mid BEACON,\ and\ S \le \min_f (RSI_f)\}$.

while $(T > D)$ {

  if $(SI > SI_{\min})$

    set $SI = \sup\{S : S \mid BEACON, S \le \min_f (RSI_f),\ and\ S < SI\}$.

  else

    return REJECT;

  recalculate T using Eq. 3.

}

return ACCEPT;

V. QoS Support for VBR traffic

Lots of multimedia applications such as voice over IP with silence suppression and video conferencing generate VBR traffic. In this section, we first present a simple scheduling algorithm which assigns fixed TXOPs to a flow. We derive an upper bound of the end-to-end delay for a VBR flow *f*, assuming that flow *f* is regulated by a leaky bucket with mean data rate ρ_f and maximum burst size σ_f. We then present a novel scheduling algorithm which assigns variant TXOPs to a flow, aiming to increase the system throughput.

A. Fixed TXOP scheduling algorithm

During each SI, the network needs to assign the time slots to all accepted flows. A simple scheme is to assign a fixed VTXOP to a VBR flow based on the flow specification and admission control policy. Obviously the bandwidth assigned to a flow should be higher than the flow's average data rate. Additional bandwidth should also be considered to cover the packet retransmission caused by transmission errors. In order not to affect the QoS of existing flows, the VTXOP of a newly accepted flow should be scheduled after those of all existing flows. The packets from a flow should be scheduled in a First-Come-First-Served (FCFS) fashion. The advantage of this simple scheduling algorithm is that we can derive an upper delay bound for a flow which conforms to a leaky bucket regulator. Hence this simple scheduling algorithm is suitable for the flows with stringent guaranteed delay requirement. The derivation of the end-to-end delay bound is based on the concept of Guaranteed Rate from [30].

We first introduce some notations. The average service rate assigned to flow *f* is ρ_f, which should be no less than the average data arrival rate of flow *f*. The *j*th packet of flow *f* is denoted by p_f^j, and assume it arrives at server *i* at time $A^i(p_f^j)$. The length of packet p_f^j is denoted by l_f^j. Let $GRC^i(p_f^j)$ denote the Guaranteed Rate clock value of packet p_f^j at server *i*, which is defined as follows.

Definition 1. The Guaranteed Rate clock value of packet p_f^j at server *i*, $GRC^i(p_f^j)$, is defined by:

$$GRC^i(p_f^j)=\begin{cases}0 & j=0\\ \max\{A^i(p_f^j),GRC^i(p_f^{j-1})\}+\dfrac{l_f^j}{\rho_f} & j>0\end{cases}.$$

Definition 2. A scheduling algorithm at server *i* belongs to class **Guaranteed Rate** (GR) for flow *f* if it guarantees that packet will be transmitted by $GRC^i(p_f^j)+\beta^i$, where β^i is a constant which depends on the scheduling algorithm and the server.

We now prove the following theorem:

Theorem 1. A scheduling algorithm belongs to the GR class if (1) in each SI, $\rho_f \cdot T_{SI}/C_i$ time period is assigned to flow *f*, where C_i is the service rate of server *i*; (2) the packets in flow *f* are scheduled in the First-Come-First-Served (FCFS) fashion; (3) all the flows are also scheduled in the FCFS fashion.

Proof.

Let $\mathrm{H}^i(p_f^j)$ denote the time at which packet p_f^j leaves server *i*, and let $\mathrm{H}^i(p_f^0)=0$. We use induction on *j* to prove $\mathrm{H}^i(p_f^j)\le GRC^i(p_f^j)+T_{SI}$.

Base case: *j* = 0

$\mathrm{H}^i(p_f^0)=0 \Rightarrow \mathrm{H}^i(p_f^0)\le GRC^i(p_f^0)+T_{SI}$.

Induction hypothesis: assume $\mathrm{H}^i(p_f^j)\le GRC^i(p_f^j)+T_{SI}$ holds for $0\le j\le t$.

Induction: we need to show that $\mathrm{H}^i(p_f^{t+1})\le GRC^i(p_f^{t+1})+T_{SI}$.

Consider packet p_f^{t+1}. There are two cases:

Case 1: $A^i(p_f^{t+1})\ge \mathrm{H}^i(p_f^t)$

In this case, packet p_f^{t+1} arrives at server *i* after the departure of packet p_f^t. Due to the FCFS principle, this packet will be served within an SI, and therefore we can have $\mathrm{H}^i(p_f^{t+1}) \le A^i(p_f^{t+1}) + T_{SI}$. Since $\max\{A^i(p_f^{t+1}), \mathrm{H}^i(p_f^t)\} = A^i(p_f^{t+1})$, we can immediately have:

$$\mathrm{H}^i(p_f^{t+1}) \le \max\{A^i(p_f^{t+1}), \mathrm{H}^i(p_f^t)\} + T_{SI}. \tag{4}$$

Case 2: $A^i(p_f^{t+1}) < \mathrm{H}^i(p_f^t)$

In this case, packet p_f^{t+1} arrives at server *i* before the departure of packet p_f^t. There are two different subcases:

(1) If packet p_f^{t+1} is transmitted in the same burst as packet p_f^t, then $\mathrm{H}^i(p_f^{t+1}) \le \mathrm{H}^i(p_f^t) + \frac{l_f^{t+1}}{C_i} \le GRC^i(p_f^t) + T_{SI} + \frac{l_f^{t+1}}{C_i}$.

Given that $GRC^i(p_f^{t+1}) = \max\{A^i(p_f^{t+1}), GRC^i(p_f^t)\} + \frac{l_f^{t+1}}{\rho_f} = GRC^i(p_f^t) + \frac{l_f^{t+1}}{\rho_f}$, we have:

$$\begin{aligned} \mathrm{H}^i(p_f^{t+1}) &\le GRC^i(p_f^{t+1}) - \frac{l_f^{t+1}}{\rho_f} + T_{SI} + \frac{l_f^{t+1}}{C_i} \\ &= GRC^i(p_f^{t+1}) + T_{SI} - \left(\frac{l_f^{t+1}}{\rho_f} - \frac{l_f^{t+1}}{C_i}\right) \\ &\le GRC^i(p_f^{t+1}) + T_{SI} \end{aligned}. \tag{5}$$

The last inequality holds because $\rho_f \le C_i$.

(2) If packet p_f^{t+1} is not in the same burst as packet p_f^t, then the only possibility is that p_f^t is the last packet of a burst, while p_f^{t+1} is the first packet of a new burst. Assume the burst of packets that includes p_f^t have totally *x* packets, denoted by p_f^{t-x+1}, p_f^{t-x+2}, …, p_f^{t-1}, p_f^t. Because the flows are scheduled by FCFS, the time interval between the schedule of p_f^{t-x+1} and p_f^{t+1} can be no longer than T_{SI}. Thus we have $\mathrm{H}^i(p_f^{t+1}) \le \mathrm{H}^i(p_f^{t-x+1}) + T_{SI}$. Furthermore, based on the assumption of $A^i(p_f^{t+1}) < \mathrm{H}^i(p_f^t)$, we can conclude that:

$$\sum_{m=t-x+1}^{t+1} l_f^m \ge \rho_f \cdot T_{SI}. \tag{6}$$

This is because otherwise p_f^{t+1} will be transmitted in the same burst as p_f^t.

From the induction hypothesis, $\mathrm{H}^i(p_f^{t-x+1}) \le GRC^i(p_f^{t-x+1}) + T_{SI}$, hence we have $\mathrm{H}^i(p_f^{t+1}) \le GRC^i(p_f^{t-x+1}) + 2T_{SI}$. (7)

On the other hand, we have

$$\begin{aligned} GRC^i(p_f^{t+1}) &\ge GRC^i(p_f^{t}) + \frac{l_f^{t+1}}{\rho_f} \ge GRC^i(p_f^{t-1}) + \frac{l_f^{t} + l_f^{t+1}}{\rho_f} \\ &\ge \cdots \ge GRC^i(p_f^{t-x+1}) + \frac{\sum_{m=t-x+1}^{t+1} l_f^m}{\rho_f} \end{aligned}, \tag{8}$$

which leads to $GRC^i(p_f^{t-x+1}) \le GRC^i(p_f^{t+1}) - \dfrac{\sum_{m=t-x+1}^{t+1} l_f^m}{\rho_f}$. (9)

From (6), (8), and (9), we can have :

$$\begin{aligned} \mathrm{H}^i(p_f^{t+1}) &\le GRC^i(p_f^{t-x+1}) + 2T_{SI} \\ &\le GRC^i(p_f^{t+1}) - \frac{\sum_{m=t-x+1}^{t+1} l_f^m}{\rho_f} + 2T_{SI} \\ &\le GRC^i(p_f^{t+1}) + T_{SI} \end{aligned}. \tag{10}$$

This completes our proof.

If flow *f* conforms to a leaky bucket with parameters (σ_f, ρ_f), following Theorem 3 in [30], the end-to-end delay of packet p_f^j, denoted by d_f^j, is bounded by:

$$d_f^j \le \frac{\sigma_f + (k-1)L_f}{\rho_f} + k \cdot T_{SI} + \sum_{i=0}^{k-1} \tau_i, \tag{11}$$

where L_f is the maximum packet size of flow *f*, and the term $\sum_{i=0}^{k-1} \tau_i$ represents the total propagation delay.

From (11), we can see that the end-to-end delay bound is mainly determined by the leaky bucket parameters (σ_f, ρ_f) and also the value of T_{SI}. Normally σ_f is specified by the user, while ρ_f and T_{SI} are assigned by the scheduler. With a smaller value of T_{SI}, poll frames will be generated more frequently which decreases the channel utilization. If the value of T_{SI} is fixed and the route is given, the value ρ_f can be determined by (11) based on the delay requirement of flow *f*. The admission control algorithm simply calculates the minimum value of ρ_f that satisfies (11), and then checks whether there is enough free bandwidth or not. If all the links have enough free bandwidth, flow *f* will be admitted and $\rho_f \cdot T_{SI} / C_i$ of time period in each SI will be allocated to flow *f* on link *i*; otherwise, flow *f* will be rejected.

B. Dynamic TXOP scheduling algorithm

It is well known that providing guaranteed QoS to VBR flows has the disadvantage of low resource utilization. If a "large" number of VBR flows are multiplexed together, it is better to adopt an admission control algorithm with statistical guarantee which takes advantage of the statistical multiplexing gain. In this section, we propose a novel scheduling algorithm and an admission control algorithm which utilizes the principle of statistical multiplexing by assigning variant VTXOPs to different VBR flows in different SIs. There are three problems to be solved: (1) How to decide the length of time used for all VBR flows in each SI? (2) In each SI, how to decide the polling sequence? (3) In each SI, how to determine the VTXOP assigned to each flow?

We first present our solution to problem (1). At different SIs, our system allocates different time periods to VBR flows, under the condition that the average time period allocated to VBR flows is $R_{VBR\_avg} \cdot T_{SI}$ in a long term and it must be no greater than $R_{VBR\_max} \cdot T_{SI}$. The time allocated to VBR flows in the i^{th} SI is denoted by $R^i_{VBR} \cdot T_{SI}$. The real time consumed by VBR flows in the i^{th} SI is denoted by $R^i_{VBR\_T} \cdot T_{SI}$, which can be measured by the QAP. If $R^i_{VBR\_T} \cdot T_{SI}$ is less than $R_{VBR\_avg} \cdot T_{SI}$, the remaining time $(R_{VBR\_avg} - R^i_{VBR\_T}) \cdot T_{SI}$ can be used for EDCA traffic; and in the next SI, VBR flows are allowed to have more transmission chances. Initially we set $R_{VBR}^1 = R_{VBR\_avg}$; afterwards, for $i > 1$, R_{VBR}^i is determined by $R_{VBR}^i = \min(R_{VBR\_max},\ i \cdot R_{VBR\_avg} - \sum_{t=1}^{i-1} R_{VBR\_T}^t)$.

Next, we introduce our solutions to problems (2) and (3). At the access link, a flow will be assigned with different VTXOP in different SIs in order to absorb the traffic burstiness. Each flow *f* with average data rate of r_f is assigned with a VTXOP with average length $VTXOP_f$ which should be no less than $r_f \cdot T_{SI} / C_0$, and a token limit σ_f is used to policy the traffic from flow *f*. If a flow is not regulated by a token, we can simply set σ_f to infinity. The value of $VTXOP_f$ depends on r_f, packet loss rate, and also the call admission control policy, which will be discussed at the end of this section.

In the i^{th} SI, before polling a wireless station[1] with a single flow *f*, the QAP needs to determine the VTXOP assigned to that station, denoted by $VTXOP_f^i$. In the first SI of flow *f*, this variable is simply set to $VTXOP_f$. After that, the value of $VTXOP_f^i$ will be dynamically changed in each SI according to the scheduling algorithm described as follows.

After flow *f* has finished its data transmission in the i^{th} SI, the real time consumed by flow *f* in this SI can be measured by the QAP, and denoted by TR_f^i, which must be no more than $VTXOP_f^i$. The token of flow *f* at the beginning of the $(i+1)^{th}$ SI, denoted by σ_f^{i+1}, is set by $\sigma_f^{i+1} = \min(\sigma_f^i - TR_f^i + VTXOP_f, \sigma_f)$. If flow *f* happens to generate a burst of traffic with a higher rate during an SI, some packets may be queued at the source node. The queue length of flow *f* at the instant that the last frame of flow *f* sent in the i^{th} SI starts to be sent is denoted by q_f^i, and is piggybacked to the QAP by the last frame. Assume there are *N* flows, the QAP sorts all the flows such that $q_{f_1}^i / r_{f_1} \le q_{f_2}^i / r_{f_2} \le \cdots \le q_{f_N}^i / r_{f_N}$. In reality, the flows

[1] For simplicity, in this section we assume each wireless station only generates one VBR flow.

under light traffic are likely have an empty queue, whilst the flows under heavy traffic may have a positive queue length. We assume that $q_{f_1}^i = \cdots = q_{f_Z}^i = 0 < q_{f_{Z+1}}^i / r_{f_{Z+1}} \leq \cdots \leq q_{f_N}^i / r_{f_N}$. In the $(i+1)^{th}$ SI, the QAP polls the stations according to the sequence of $f_1, f_2, ..., f_N$. The rationale behind this heuristic is that, the queue length can be used to estimate the state of a flow. A large value of q_f^i / r_f implies that the flow is under heavy traffic, and we shall assign more VTXOP to it if possible. For flow *f* with $q_f^i = 0$, we expect it to be under light traffic, and hence we set $VTXOP_f^{i+1} = VTXOP_f$. In fact, we also expect that such a flow *f* will not use up $VTXOP_f$, and the saved time can be shared by the remaining flows.

After serving the first Z flows, the residual time period is $R_{VBR}^{i+1} \cdot T_{SI} - \sum_{t=1}^{Z} TR_{f_t}^{i+1}$. The last problem is how to distribute the residual time to the remaining $N-Z$ VBR flows. Our scheduling algorithm determines $VTXOP_{f_j}^{i+1}$ for flow f_j after flow f_{j-1} has been served and the value of $TR_{f_{j-1}}^{i+1}$ has been measured: $VTXOP_{f_j}^{i+1} = \min(VTXOP_{f_j} + \alpha_{f_j}^{i+1}, \sigma_{f_j}^{i+1})$, where

$$\alpha_{f_j}^{i+1} = \frac{q_{f_j}^i / r_{f_j}}{\sum_{t=j}^{N} (q_{f_t}^i / r_{f_t})} (R_{VBR}^{i+1} \cdot T_{SI} - \sum_{t=1}^{j-1} TR_{f_t}^{i+1} - \sum_{t=j}^{N} VTXOP_{f_t}).$$

The remaining issue is how to determine the value of $VTXOP_f$ for flow *f* with average data rate r_f. To simplify our discussion, we assume an error-free transmission system. In reality, to compensate for the transmission errors, more resources should be reserved for each flow which can be used for frame retransmission or error correction codes. If the flow has no delay requirement, we can set $VTXOP_f$ as r_f to maximize the statistical multiplexing gain. In reality, lots of multimedia applications have some delay requirement, however. Equivalent bandwidth is a well-studied technique used for admission control to provide statistical performance guarantee for VBR flows. Normally the equivalent bandwidth of a flow *f*, denoted by E_f, is calculated based on the flow parameters such as mean rate, peak rate, burst size, and also the flow QoS requirement such as the packet loss rate and/or delay bound. Lots of existing techniques can be used for such calculations [31, 32]. A new flow f can be admitted if on every link i along the route, $E_f + \sum_{l:\ all\ existing\ flows} E_l < R_{VBR\_avg} \cdot C_i$. If flow f were admitted, we can set $VTXOP_f$ to E_f. The difficulty of applying the concept of equivalent bandwidth into real applications lies in the fact that it is extremely difficult, if not impossible, to precisely model the traffic flow of a VBR application.

Similarly, a backbone wireless mesh router can use the dynamic TXOP assignment to determine the time period used for VBR traffic in each SI.

VI. Simulation Studies

In order to evaluate the proposed scheduling algorithms, we have developed an event-driven simulator using C++. We simulate a six-hop tandem topology which contains six wireless mesh routers. We assume that all the mobile users are attached at the first wireless router, and all the traffics are destined to the last wireless router, which is the Internet gateway. IEEE 802.11g is used at the first wireless link, and IEEE 802.11a is used at all the backbone links, both working at 54Mbps. Furthermore, we assume that two adjacent links use non-overlapping channels. We set the time of sending a QoS CF-POLL/QoS NULL/ACK frame to 32μs, and a SIFS takes 10μs.

A. Performance evaluation of CBR flows

We first evaluate the delay performance of CBR traffic by simulating the traffic generated by G.711 voice codec. The payload size is 160 bytes, the IP/UDP/RTP header length is 40 bytes, and the packet inter-arrival period is 20 ms. We set T_{SI} =20 ms, and R_{CBR} =20%, i.e., in each SI, 4 ms are reserved for CBR flows. The total time used to deliver one packet from the source to the AP is 160μs, which includes the time of a poll frame, a data frame, an ACK frame, and three SIFSs. It is obvious that the overhead is very high, and some work has been done to improve the channel utilization for such voice application [33]. If we assume error free transmission, 25 flows can be admitted at a wireless access link (with 20% capacity dedicated to voice application) based on our admission control scheme. From Eq. (2), the packet delay of the CBR flow is bounded by 143 ms. Our simulation results are shown in Fig. 9. One observation is that, all the maximum delays are much smaller than the theoretical bound. The main reason is that all the six wireless mesh routers are sending out polling frames independently, and hence the worst case is very unlikely to happen. The average delays are around 80 ms, which is good enough to support voice applications. If the number of wireless hops grows, we can decrease the value of T_{SI} to decrease the delay.

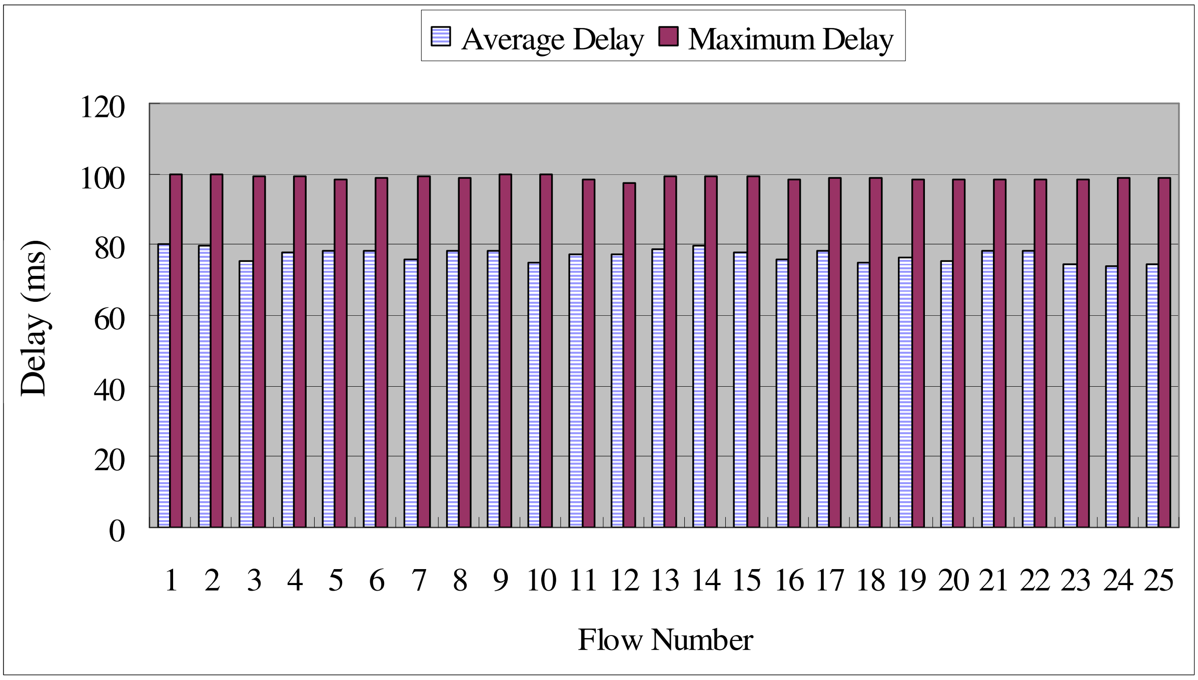

Fig. 9: Delay performance of 25 CBR flows

B. Performance evaluation of fixed TXOP assignment for VBR flows

In this section, we evaluate the performance of fixed-TXOP scheduling scheme for VBR flows which conform to leaky bucket regulators. Assume flow *f* conforms to a leaky bucket with parameters (σ_f, ρ_f). In our simulation, the packets of flow *f* are generated at the source node according to the worst-case traffic model: σ_f of data are generated periodically with a time interval of σ_f / ρ_f. We consider two types of flows: Type I with traffic parameters (256 Kbits, 512 Kbps), Type II with traffic parameters (1024 Kbits, 512 Kbps). For simplicity, we assume a constant packet payload size of 640 bytes. In all the simulations in this section, we set R_{VBR}=50% and $T_{SI} = 20$ ms.

In the first scenario, there are 13 Type I flows with even flow numbers and 13 Type II flows with odd flow number. Every flow is assigned with a bandwidth of 512 Kbps. From Eq. (11), the delay of Type I flow is bounded by 626 ms, and the delay of Type II flow is bounded by 2176 ms. The simulation results of the maximum packet delay as well as the average packet delay are shown in Fig. 10. We can see that the maximum packet delays are all below the theoretical bound. Another observation is that the average delay of Type II flows is much longer than that of Type I flows, albeit both types have the same average data rate. This is mainly because that Type II flows are much burstier than Type I flows.

In the second scenario, we simulate 12 Type I flows, each of which is assigned with a bandwidth of 512 Kbps, and 4 Type II flows, each of which is assigned with a bandwidth of 2048 Kbps. The objective of assigning 2048 Kbps of bandwidth to Type II flows is to lower down the delay bound to 626 ms. The simulation results are shown in Fig. 11. All the maximum delays are below the theoretical bound. We can also observe from the figure that the delay performances of both types of flows are almost the same now. This confirms the effectiveness of improving the delay performance by allocating more bandwidth. Nevertheless, the cost of guaranteed delay bound is the low channel utilization: only 4 Type II flows can be admitted into the network.

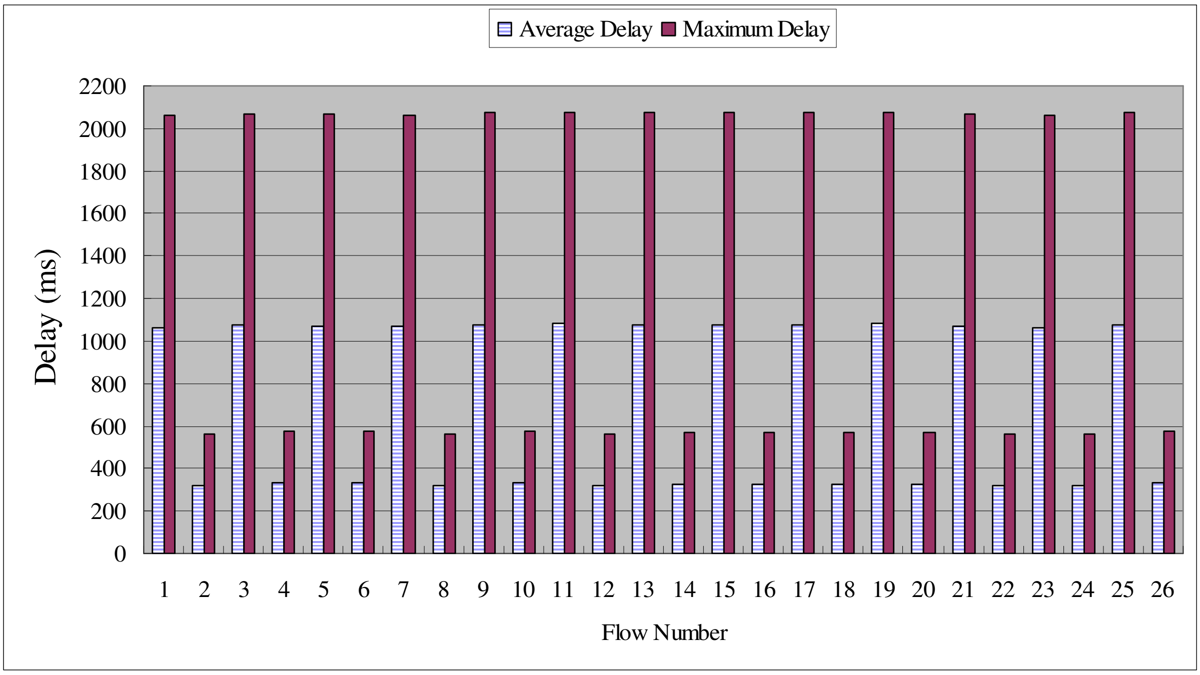

Fig. 10: Delay performance of 13 Type I flows and 13 Type II flows: each flow is assigned with 512 Kbps

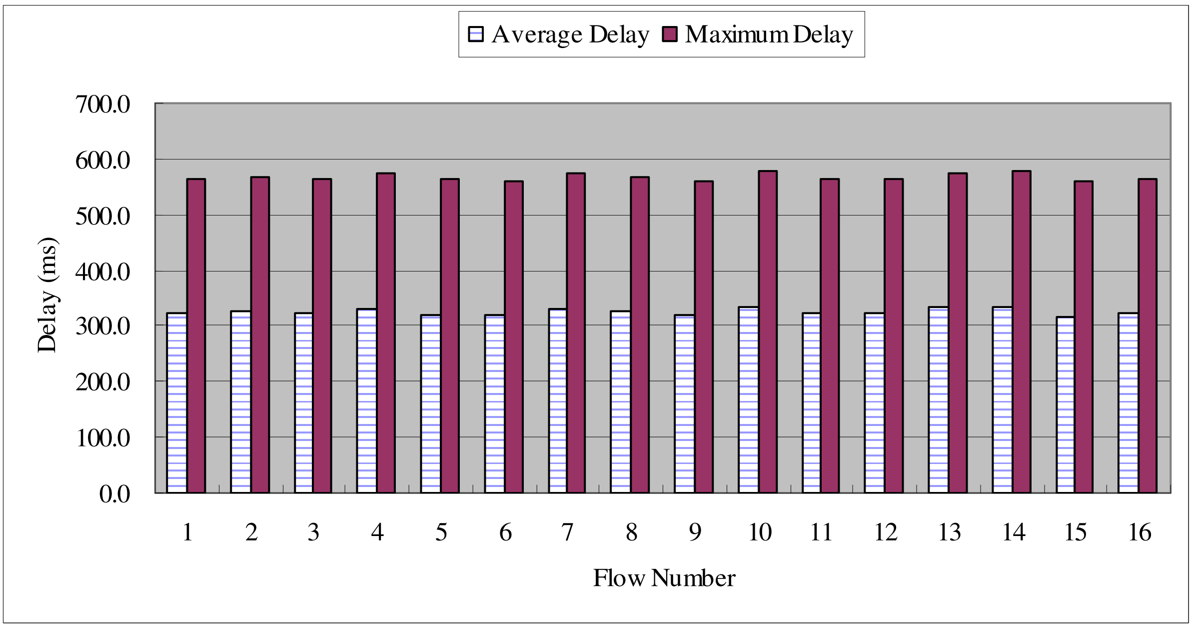

Fig. 11: Delay performance of 13 Type I flows and 4 Type II flows:
each Type I flow is assigned with 512 Kbps, each Type II flow is assigned with 2048 Kbps

C. Performance evaluation of dynamic TXOP scheduling for VBR flows

In this section, we present the simulation results of the dynamic TXOP scheduling algorithms for VBR flows. For all related simulations, we set T_{SI} = 20 ms, R_{VBR_avg} = 50%, and R_{VBR_max} = 70%.

We first conduct simulations on VBR flows regulated by leaky-buckets. For comparison purpose, we simulate 26 flows which contain 13 Type I flows with odd flow number and 13 Type II flows with even flow number, where “Type 1” and “Type 2” are defined in Section VI.B. We set $VTXOP_f$ to 512 Kbps, which achieves the maximum multiplexing gain since the average data rate of each flow equals 512 Kbps. The simulation results of average packet delays and maximum packet delays of all flows are shown in Fig. 12. If we compare Fig. 12 with Fig. 10, we can observe that the average delay of Type I flows is decreased by 60%, while the average delay of Type II flows is decreased by 80%. This implies that the dynamic TXOP scheduling algorithm outperforms the fixed TXOP scheduling algorithm significantly in terms of delay performance.

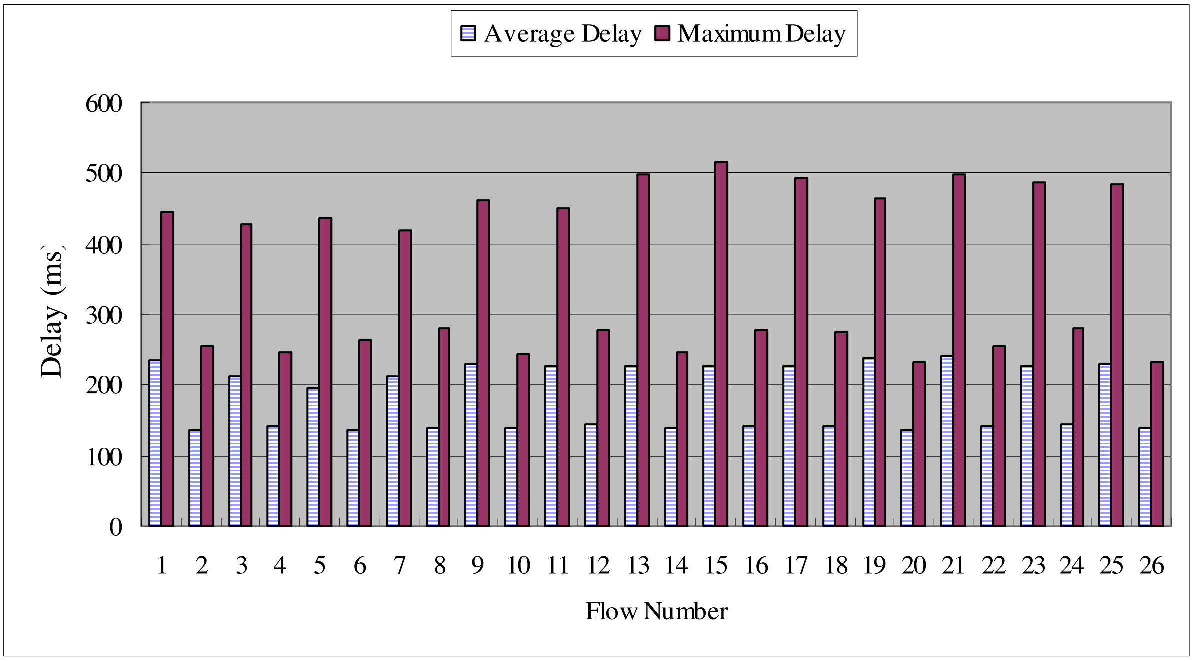

Fig. 12: Delay performance of 13 Type I flows and 13 Type II flows: dynamic TXOP scheduling

We also conduct simulations for VBR flows without leaky-bucket regulation. We adopt an ON/OFF traffic model in which the ON and OFF periods are exponentially distributed. The average time of on period is denoted by t_{avg_on}, and the average time of off period is denoted by t_{avg_off}. During the ON periods, the packets with fixed length 640 bytes arrive at a constant data rate r_{on}. It is easy to see that the average flow data rate r is given by $r = r_{on} \cdot t_{avg\_on} / (t_{avg\_on} + t_{avg\_off})$. We conduct two sets of simulations: (1) t_{avg_on} = t_{avg_off} = 500 ms, r = 512 kbps, r_{on} = 1024 kbps; (2) t_{avg_on} = 200 ms, t_{avg_off} = 800 ms, r = 512 kbps, r_{on} = 2560 kbps.

In the first set of simulations, the average time of ON period is the same as that of OFF period. This configuration simulates flows with modest burstiness. We set $VTXOP_f$ to 512 kbps, and hence at most 26 flows can be admitted at the access link. We conduct simulations with 26 simultaneous flows, i.e., 100% utilization of the access link. In Fig. 13, we plot the delay performance of 4 flows out of 26 flows that include the flow with the best average delay performance, the flow with the worst delay performance, and another two flows in the middle. From the figure, we can see that more than 98% of the packets have the delay below 100ms, which is good enough for real-time multimedia applications. For comparison purpose, in Fig. 14 we show the delay performance of the same traffic but with the fixed TXOP scheduling algorithm. When fixed TXOP scheduling algorithm is used, very large packet delays up to several seconds have been observed. This further validates the effectiveness of the dynamic TXOP scheduling algorithm.

In the second set of simulations, the average time of ON period is set to 200ms and the average time OFF period is set to 800ms. The average data rate is still 512 Kbps, so the maximum number of flows is still 26. This configuration simulates flows with large burstiness. We set $VTXOP_f$ to 512 kbps, and evaluate the delay performance for different number of flows. The simulation results are shown in Fig. 15. With 25 or 26 flows, i.e., 95% - 100% of channel utilization, the delay performance is not satisfactory due to the large burstiness of the traffic. But with 24 flows or less, the delay performance is very promising: more than 95% of packets have a delay less than 250 ms, which is acceptable for a large range of multimedia applications.

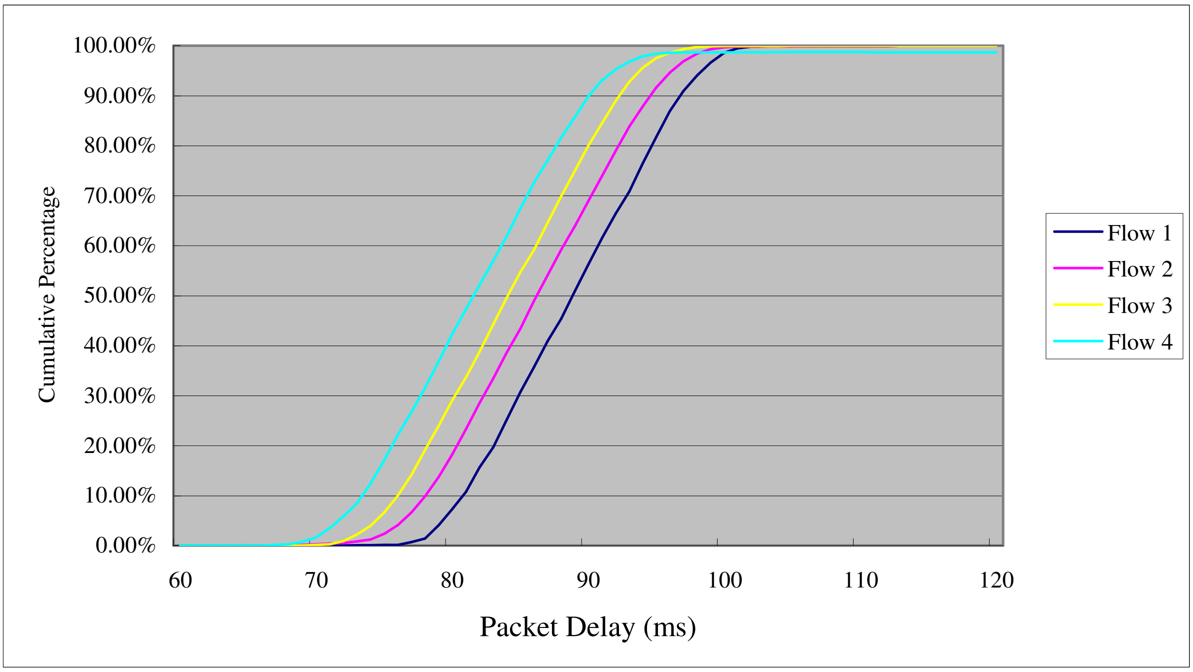

Fig. 13: Delay performance of dynamic TXOP scheduling with 26 flows

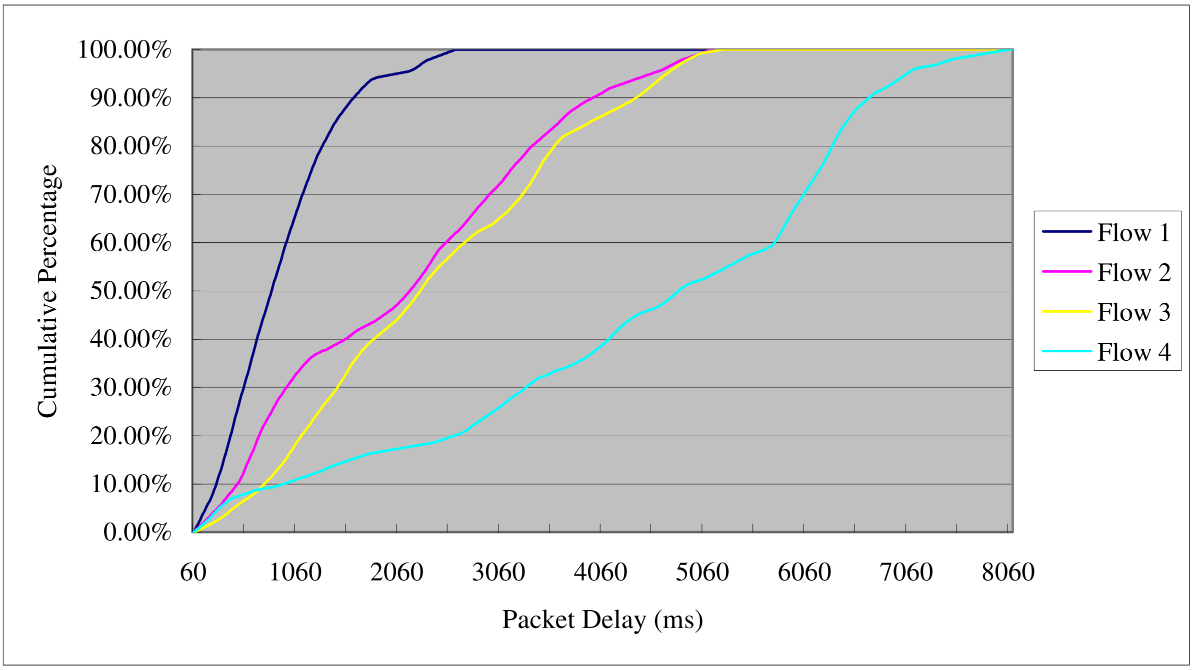

Fig. 14: Delay performance of fixed TXOP scheduling with 26 flows

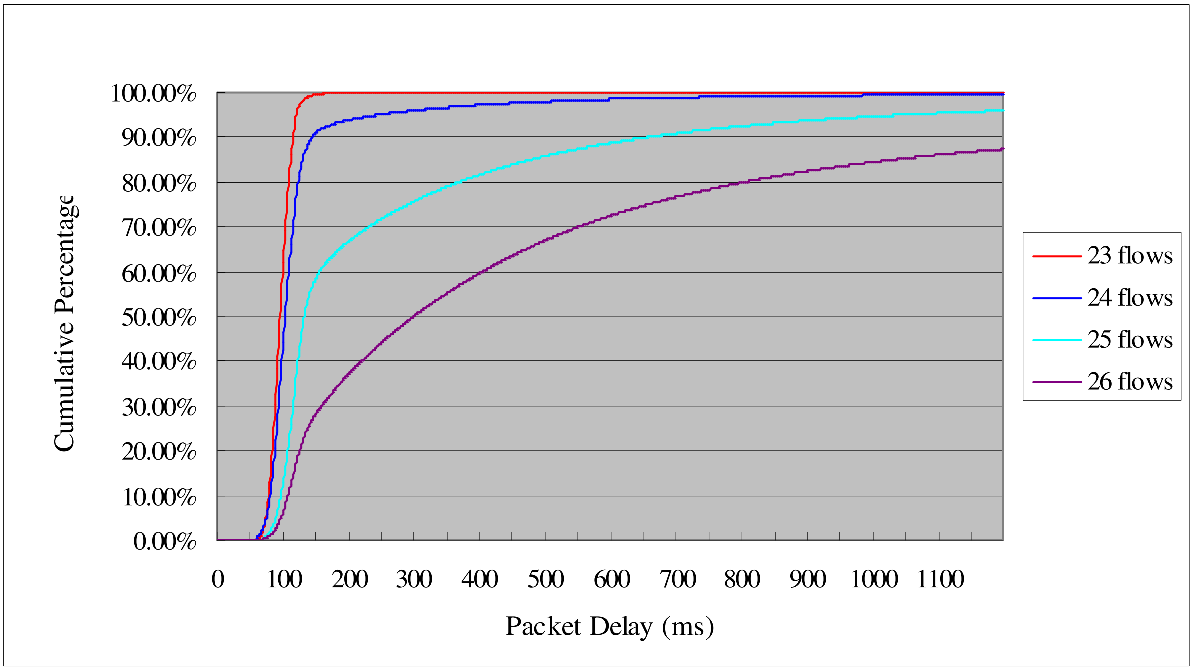

Fig. 15: Delay performance of dynamic TXOP scheduling with different flows

VII. Conclusions

In this paper, we have proposed a framework for provisioning of parameterized QoS in IEEE 802.11e based wireless mesh networks. Our framework can support CBR traffic with strict delay guarantee, and also VBR traffic w/o delay guarantee. We have derived an upper delay bound for CBR traffics, which can be used in the admission control algorithm for CBR flows. For VBR flows with strict delay requirement, we proposed a fixed TXOP scheduling algorithm and derived an upper delay bound for VBR flows that conform to leaky-bucket regulators. The guaranteed delay is achieved at the expense of allocating more bandwidth to the flows, however. For VBR flows without strict delay requirement, we proposed a dynamic TXOP scheduling algorithm. Through extensive simulation studies, we have shown that our dynamic TXOP scheduling algorithm outperforms the fixed TXOP scheduling algorithm for VBR flows regulated by leaky-buckets. More importantly, for VBR flows without leaky-bucket regulators, the dynamic TXOP scheduling algorithm has been shown to be very effective to achieve high channel utilization ratio while keeping an acceptable delay performance.

Acknowledgement

This work was partially supported by Hong Kong RGC under CERG grant HKBU 210406, a Central Allocation grant HKBU 1/05C, and Hong Kong Baptist University under grant FRG/06-07/II-69. We also thank Mr. Yong Yan for implementing the simulations.

References

[1] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless Mesh Networks: a Survey," Computer Networks Journal (Elsevier), vol. 47, no. 4, pp. 445-487, March 2005.

[2] I. F. Akyildiz and X. Wang, "A Survey on Wireless Mesh Networks," IEEE Communications Magazine, vol. 43, no. 9, s23-s30, Sept. 2005.

[3] R. van Drunen, D. van Gulik, J. Koolhaas, H. Schuurmans, and M. Vijn, "Building a Wireless Community Network in the Netherlands," in Proc. of USENIX 2003 Annual Technical Conference, FREENIX Track, pp. 219-230, 2003.

[4] R. Karrer, A. Sabharwal, and E. Knightly, "Enabling Large-scale Wireless Broadband: the Case for TAPs," in Proc. of HotNets, Cambridge, MA, 2003.

[5] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, "Link Level Measurements From an 802.11b Mesh Network," in Proc. of ACM Sigcomm, pp. 121-131, USA, August 2004.

[6] J. Bicket, S. Biswas, D. Aguayo, and R. Morris, “Architecture and Evaluation of the MIT Roofnet Mesh Network,” in Proceedings of ACM MobiCom, Cologne, Germany, August 2005.

[7] R. Draves, J. Padhye, and B. Zill, “Comparison of Routing Metrics for Static Multi-hop Wireless Networks,” in Proc. of ACM Sigcomm, pp. 133-144, USA, August 2004.

[8] R. Draves, J. Padhye, and B. Zill, “Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks,” in Proc. of ACM Mobicom, pp. 114-128, September 2004.

[9] V. Gambiroza, B. Sadeghi, and E. Knightly, “End-to-End Performance and Fairness in Multihop Wireless Backhaul Networks,” in Proc. of ACM Mobicom, September 2004.

[10] J. Camp, J. Robinson, C. Steger, and E. Knightly, “Measurement Driven Deployment of a Two-Tier Urban Mesh Access Network,” in Proc. of ACM MobiSys’06, 2006.

[11] J. Eriksson, S. Agarwal, P. Bahl, and J. Padhye, “Feasibility Study of Mesh Networks for All-Wireless Offices,” in Proc. of ACM MobiSys’06, pp 69-82, 2006.

[12] P. Bahl, A. Adya, J. Padhye, and A. Walman, “Reconsidering Wireless Systems with Multiple Radios,” ACM SIGCOMM Computer and Communications Review, Vol 34, No. 5, pp 39-46, 2004.

[13] A. Raniwala, K. Gopalan, and Tzi-cker Chiueh, “Centralized Channel Assignment and Routing Algorithms for Multi-channel Wireless Mesh Networks,” ACM SIGMOBILE Mobile Computing and Communications Review, 8(2):50-65, 2004.

[14] A. Raniwala and T. cker Chiueh, “Architecture and Algorithms for an IEEE 802.11-based Multi-channel Wireless Mesh Network,” in Proc. of IEEE Infocom’05, 2005.

[15] P. Kyasanur and N. Vaidya, “Capacity of Multi-channel Wireless Networks: Impact of Number of Channels and Interfaces,” in Proc. of ACM MobiCom’05, 2005.

[16] P. Kyasanur, J. So, C. Chereddi, and N. Vaidya, “Multi-Channel Mesh Networks: Challenges and Protocols”, in IEEE Wireless Communications, April 2006.

[17] J. Shi, T. Salonidis, and E. Knightly, “Starvation Mitigation Through MultiChannel Coordination in CSMA Multihop Wireless Networks,” in Proc. of ACM MobiHoc’06, 2006.

[18] H. Wu, F. Yang, K. Tan, J. Chen, Q. Zhang, and Z. Zhang, “Distributed Channel Assignment and Routing in Multiradio Multichannel Multihop Wireless Networks,” IEEE Journal on Selected Areas in Communications, Vol. 24, No. 11, pp 1972-1983, November 2006.

[19] IEEE 802.11 part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, August 1999.

[20] IEEE 802.11 part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-speed Physical Layer Extension in the 2.4 GHz Band, September 1999.

[21] IEEE 802.11 part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-speed Physical Layer in the 5 GHz Band, September 1999.

[22] IEEE 802.11 part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications, Amendment 4: Further higher data rate extension in the 2.4GHz Band, June 2003.

[23] N. Ramos, D. Panigrahi, and S. Dey, “Quality of Service Provisioning in 802.11e Networks: Challenges, Approaches, and Future Directions,” IEEE Network, pp.14-20, July/August 2005.

[24] H. Jiang, W. Zhuang, X. Shen, A. Abdrabou, and P. Wang, “Differentiated Services for Wireless Mesh Backbone,” IEEE Communications Magazine, vol. 44, no. 7, pp. 113-119, July 2006.

[25] IEEE 802.11 part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications, Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements, November 2005.

[26] T. B. Reddy, I. Karthigeyan, B. S. Manoj, and C. Siva Ram Murthy, “Quality of Service Provisioning in Ad hoc Wireless Networks: A Survey of Issues and Solutions,” Ad Hoc Networks Journal, vol. 4, no. 1, pp. 83-124, January 2006.

[27] W. F. Fan, D. Gao, D. H. K. Tsang, and B. Bensaou, “Admission Control for Variable Bit Rate Traffic in IEEE 802.11e WLANs,” in Proc. of IEEE LANMAN’04, pp.61-66, 2004.

[28] D. Gao, J. Cai, and K. N. Ngan, “Admission Control in IEEE 802.11e Wireless LANs,” IEEE Network, pp.6-13, July/August 2005.

[29] L. Hanzo II. and R. Tafazolli, “A Survey of QoS Routing Solutions for Mobile Ad Hoc Networks,” IEEE Communications Surveys and Tutorials, 2nd Quarter, 2007.

[30] P. Goyal, S. Lam, and H. M. Vin, “Determining End-to-end Delay Bounds in Heterogeneous Networks,” Springer Multimedia Systems, vol. 5, pp. 157-163, 1997.

[31] R. Guerin, H. Ahmadi, and M. Nagshineh, “Equivalent Capacity and Its Application to Bandwidth Allocation in High-Speed Networks,” IEEE Journal on Selected Areas in Communications, Vol. 9, No. 7, pp. 968-981, September 1991.

[32] A. Elwalid, D. Mitra, and R. H. Wentworth, “A New Approach for Allocating Buffers and Bandwidth to Heterogeneous, Regulated Traffic in an ATM Node,” IEEE Journal on Selected Areas in Communications, Vol. 13, No. 6, pp. 1115-1127, August 1995.

[33] P. Wang, H. Jiang, and W. Zhuang, “IEEE 802.11e Enhancement for Voice Service,” IEEE Wireless Communications, pp. 30-35, February 2006.