Doctoral Thesis Proposal

Improving Quality of Service for VoIP Traffic in IEEE 802.11 Wireless Networks

Sangho Shin
Department of Computer Science
Columbia University
sangho@cs.columbia.edu

December 13, 2006

Abstract

Even though the usage of VoIP traffic in wireless networks is increasing due to widely deployed wireless networks, the Quality of Service (QoS) for VoIP traffic in wireless networks is not satisfactory. The biggest problem in VoIP traffic in wireless networks is the significant fluctuation of the delay due to the unreliability and the limited capacity of wireless networks: the delay of VoIP traffic fluctuates significantly even when the number of VoIP source is below the capacity, and the overall delay of all VoIP flows drastically increases when the number of VoIP sources approaches the capacity, due to the characteristics of CSMA/CA.

I propose a new packet scheduling algorithm called Adaptive Priority Control (APC) to alleviates the delay fluctuation and improve the capacity, and a novel call admission control with the Queue size Prediction using Computation of Additional Transmission (QP-CAT) to avoid admitting an excessive number of simultaneous calls. APC adopts the optimal priority of the downlink traffic adaptively to the channel condition in order to balance the downlink and uplink delay, which minimizes the fluctuation of delay and improves the capacity for VoIP traffic. In QP-CAT, the AP can accurately estimate the effect of a new VoIP flow on the delay of all VoIP flows by predicting queue size of the AP, before the new VoIP flow is actually admitted, and it can make exact admission decisions for new calls.

Contents

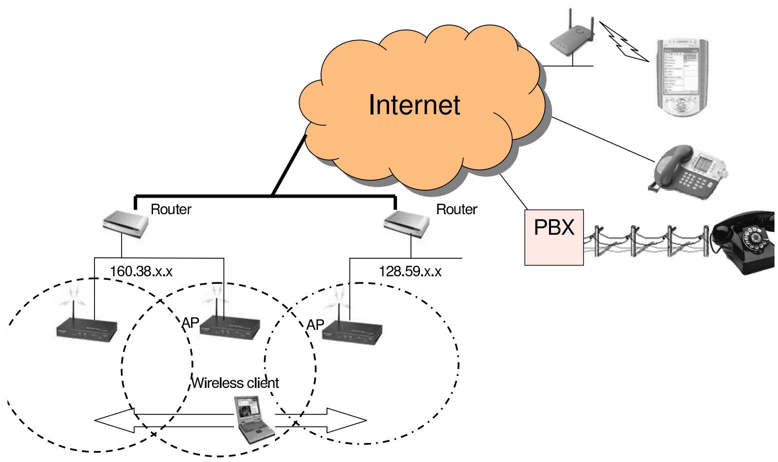


Figure 1: The framework of the VoIP traffic over IEEE 802.11 wireless networks

1 Introduction

For the past few years, IEEE 802.11 wireless networks (WiFi) have been popular due to their cheap cost and easy deployment, but the Quality of Service (QoS) of real time service such as VoIP traffic in wireless networks does not meet the growth. We can see many public APs in many coffee shops, air ports and shopping malls. Also, recently, many cities have been deploying freely accessible APs in the streets and parks so that people can use wireless networks free even without any subscription to the service, which enables people to connect to the Internet anywhere and anytime. Due to the growth of these wireless networks, the usage of VoIP service through wireless networks has been increasing. Many companies produce VoIP wireless phones or PDAs that support both the cellular and 802.11 wireless networks, and very recently a major cellular phone service provider started a service plan that allows users to call through both cellular and WiFi networks. Therefore, an explosive increase of VoIP users in wireless networks is expected in the near future, causing a lot of QoS problems.

1.1 QoS problems for VoIP traffic in wireless networks

Fig. 1 shows the framework of the VoIP in IEEE 802.11 wireless networks. Wireless clients associate with an AP and exchange VoIP packets via the AP connected to Internet through routers. The coverage range of an AP is limited and wireless clients need to change the AP when they move out of the range of the currently associated AP. Here, the first QoS problem, network disruption due to layer 2 handoffs, occurs.

The handoff happens when client stations move between two Access Points (APs). When the handoff happens in the same subnet, a layer 2 handoff occurs, and if the subnet changes due to layer 2 handoffs, a layer 3 handoff also needs to be performed. Typically, the handoff takes sufficiently long to disrupt the connection, which is undesirable for VoIP traffic. Therefore, the handoff needs to be performed seamlessly for better QoS of VoIP traffic.

Another problem of VoIP traffic in wireless networks is the limited capacity for VoIP traffic. Even though the maximum transmission rate is expanded to 54 Mb/s in 802.11a/g, the actual available bandwidth is less than a half due to the overheads and inefficiency of 802.11 MAC protocol

[18], the backoff time and the long preamble comparing to the small VoIP payload, and the waste of bandwidth from collisions and the recovery procedure. Also, the bandwidth needs to be shared with other best effort traffic such as P2P and HTTP traffic. Therefore, the capacity for the VoIP traffic needs to be increased by improving the MAC protocol.

The next problem is the imbalanced downlink and uplink delay. CSMA/CA gives the same opportunity to access medium to all nodes including the AP, which results in an unfair resource distribution between the uplink and downlink. Typically, the AP has far more packets to transmit than each wireless client, but it has the same chance to transmit packets, which causes the downlink delay to increase. The gap between the uplink and downlink delay becomes significant when the channel becomes congested, and the imbalance of the delay degrades the QoS for VoIP traffic and reduces the capacity for VoIP traffic. Therefore, a new fair packet scheduling method to balance the uplink and downlink delay is required.

The final problem is that when the number of flows in a BSS (Basic Service Set) exceeds the capacity of the channel, the overall QoS of all flows drastically deteriorates. In VoIP traffic, even when the delay of all flows is very small, a new call can cause a significant increase of the delay of all flows if the number of VoIP flows exceeds the capacity. Therefore, an efficient call admission control is necessary to protect the QoS of the existing VoIP flows.

As part of the earlier part of my thesis research, I have already proposed some solutions to the first three problems, which will be mentioned in the next section. In this proposal, I focus on the last two problems, the fairness between the uplink and downlink and the call admission control. Regarding the imbalance problem, I propose a packet scheduling technique at the AP called 'Adaptive Priority Control (APC)', where the AP computes the optimal priority of the AP to the channel condition and traffic load in order to balance the uplink and downlink delay, and the AP transmits packets contention free according to the computed priority. As the result, APC minimize the fluctuation of delay and also improve the capacity for the VoIP traffic. Also, I propose a novel call admission control method called 'Queue size Prediction with Computation of Additional Transmission (QP-CAT)', where the AP predicts its queue size of the case when a new VoIP call is admitted, by monitoring the current transmission behavior. Using QP-CAT, the AP can make a precise and efficient admission decision even before new calls are admitted.

1.2 Summary of earlier research

1.2.1 Reducing layer-2 handoff delay

Fig. 2 shows the handoff procedure defined in the IEEE 802.11 standard [12]. When the signal strength of the current AP drops below a threshold, clients need to scan APs by transmitting probe request frames. Then, according to the signal strength or other metrics in probe response frames APs sent, the next AP is chosen and the client associates with the new AP. According to Misra et al. [1] and our measurements [26], the total layer 2 handoff delay takes 300 ms to more than 1 second, and the probing delay is responsible for more than 90% of total layer 2 handoff delay. For this reason, a lot of efforts to reduce the scanning delay have been made to achieve the seamless handoff, but most of the solutions require changes of infrastructure, or the improvment does not meet the criteria for VoIP traffic.

Therefore, I have proposed a practical and efficient layer 2 handoff algorithm called 'Selective Scanning and Caching' [26], which allows clients to achieve seamless handoff in most cases. In

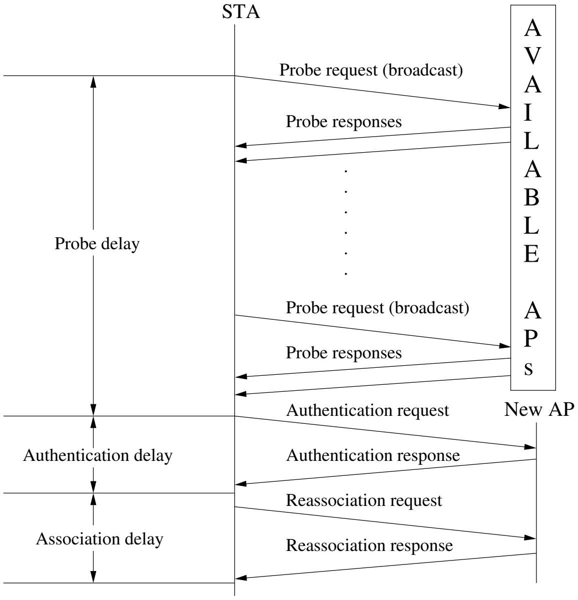


Figure 2: MAC layer handoff procedure using active scanning

Selective Scanning, clients scan non-overlapping channels first, for example, channel 1, 6 and 11 in IEEE 802.11b, reducing the scanning time to below 100 ms. Further, clients store the scanned AP information such as the channel number and the MAC address in a cache so that they can perform handoff using the information without scanning, which reduces the handoff time below 4 ms. When we use wireless nodes or VoIP phones in the same places like home or offices, we can achieve seamless handoffs in most of the cases using our approach. Furthermore, our approach requires changes in wireless card drivers of clients only, which makes it practically deployable.

1.2.2 Fast layer-3 handoff

When the subnet changes due to layer-2 handoffs, a layer-3 handoff needs to be performed. Layer-3 handoffs involve new IP address acquisition through Dynamic Host Configuration Protocol (DHCP) [8] and update of the network configuration. Layer 3 handoff is largely composed of three steps, subnet change detection, new IP address acquisition and network configuration update. Because there is no standard subnet change detection mechanism in IPv4, many operating systems realize the subnet change after a lot of packet loss, which typically takes more than a minute. Acquiring a new IP address through DHCP also takes more than one second because of Duplicate Address Detection (DAD) in DHCP servers. For these reasons, the layer 3 handoff takes a few seconds to more than a minute. Mobile IP was proposed to solve the problem a few years ago, but it is not widely deployed due to many disadvantages and significant overhead [3]. Therefore, I proposed a practical layer 3 handoff algorithm [9].

Our layer-3 handoff approach is composed of two phases, the subnet discovery using a bogus DHCP request, and the fast IP address acquisition using a temporary IP address. When a layer-2 handoff finishes, clients broadcast a bogus DHCP request with an IP address of 127.0.0.1, and then the DHCP server of the subnet responds with NACK because the IP address cannot be assigned to the clients. The NACK packet contains the IP address of the DHCP server or the relay agent, which

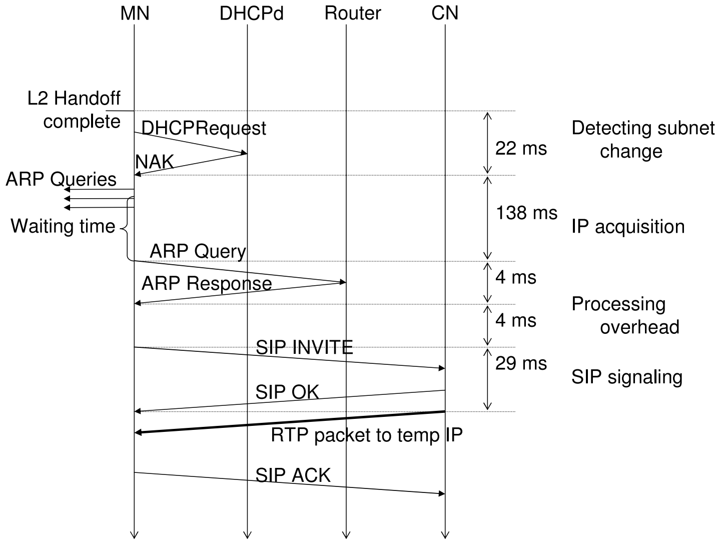


Figure 3: layer 3 handoff procedure using the temporary IP address approach

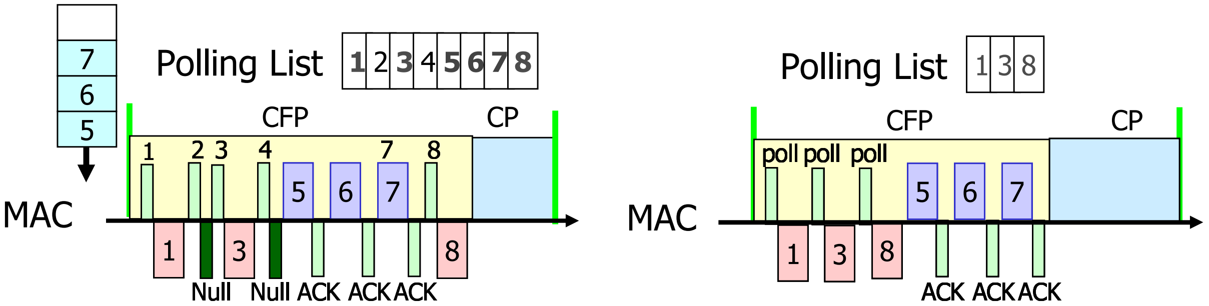


Figure 4: Polling list in PCF and DPCF

is installed per subnet, and clients can detect a subnet change by comparing it with the old DHCP server or relay agent IP address. When subnet change is detected, clients start to search for unused IP addresses in the new subnet by transmitting multiple ICMP requests to candidate IP addresses in parallel and then request a new IP address from the DHCP server. After waiting for a certain amount of time, clients pick a non-responded IP address as a temporary IP address. The client updates the application layer session with the temporary IP address and continues the communication until it gets a new IP address. Using this approach, the total handoff delay decreases from more than a second to 300 ms, and further it can be reduced to less than 10 ms when clients come back to the previous subnet before the old IP address lease expires, by reusing the old IP address without scanning temporary IP addresses. Fig. 3 shows the whole layer 3 handoff procedure when SIP is used as the application layer protocol.

1.2.3 Dynamic PCF

In order to improve the capacity for VoIP traffic, I proposed the Dynamic Pointed Coordination Function (DPCF) [18], which is based on 802.11 standard PCF. Even though the PCF is part of the standard, it is optional and not implemented in most commercial wireless cards because of many problems. The biggest problem is the waste of bandwidth from polls and Null Function frames that are transmitted even when clients do not have packets to transmit. In particular, it is a big disadvantage in VBR VoIP traffic because when a party talks, the other party usually does not talk. In DPCF, to eliminate the waste, a dynamic polling list, which contains only the talking nodes, was introduced, and the AP does not transmit polls to silent nodes (Fig. 4). Using the DPCF, the capacity for 64 kb/s VoIP traffic is improved by 32%.

1.3 Related work

1.3.1 Fairness between uplink and downlink delay

Many papers have studied the fairness among wireless nodes ([7] [4] [2] [22]), and some papers have also considered the fairness between the AP and wireless nodes ([19], [17]). However, they have focused only on the throughput fairness and failed to consider the balance of the end-to-end delay, which is more important for VoIP traffic. According to our experiments, Jain's Fairness Index [16], which is generally 1 when every node shares the throughput equally and was computed including all wireless and Ethernet nodes, is very close to 1 even when uplink and downlink delay are significantly unbalanced.

The following papers considered the balance of uplink and downlink delay. Wang et al. [28] introduced the New Fair MAC to improve the fairness and delay of VoIP traffic. When a wireless client wins a transmission opportunity, it is allowed to transmit a burst of packets instead of a packet. However, allowing clients to transmit a burst of packets does not help much for VoIP traffic because only one VoIP packet is sent every packetization interval. Also, for fairness between stations, they introduced the Max Transmission Time (MTT). Considering that packetization intervals are usually 10 ms to 40 ms and the uplink delay is very low even when the number of VoIP nodes exceeds the capacity (Fig. 6), only one packet will be sent during the MTT as in DCF, and for this reason, the delay decreased by only a few milliseconds.

Casetti et al. [6] improved the fairness for VoIP between nodes and the AP in IEEE 802.11e Enhanced Distributed Channel Access (EDCA) by differentiating frames based on traffic type and also direction. They found the optimal Contention Window (CW) values for the best fairness and throughput of VoIP traffic via simulation, and improved the capacity of VoIP by around 15%. However, they tested the optimal CW values with only one type of VoIP traffic and failed to show that the optimal value works for other types. In our study, we found that the optimal parameters should be changed according to the number of VoIP nodes and type of VoIP traffic. Also, it was discovered that changing CW values to control the priority of frames has limitation. In our approach, the parameter for fairness changes adaptively according to the network condition and we use contention free transmission approach, which does not have such limitations.

1.3.2 Call admission control

Yang Xiao et al. [29] proposed an admission control algorithm based on 802.11e EDCA [13]. The AP computes the available bandwidth using TXOP[1] of current admitted flow and announces it to

[1] An interval of time when a particular quality of service (QoS) station (QSTA) has the right to initiate frame exchange sequences onto the wireless medium [13].

clients. While this method guarantees a certain amount of bandwidth, it does not guarantee low delay. Because of it, this approach is mainly applicable to video traffic.

Pong et al. [23] estimate the available bandwidth with an analytical model. When a client requests a certain bandwidth, the AP computes the collision probability by passively monitoring the channel, and computes the available bandwidth changing the CW/TXOP and check if the requested bandwidth fits. This method shares the same problem as [29] in that it guarantees the bandwidth only. Also the assumption of the analytical method that channels are always saturated is far from true in real environments.

Sachin et al. [10] proposed a new metric for admission control, the channel utilization estimate (CUE), which is the fraction of time per time unit needed to transmit the flow over the network. The CUE is computed per flow using the average transmission rate measured for a short time and the average backoff that measured at the AP, and total CUE is calculated by summing up the CUEs of all flows. Assuming that 15% of the total network capacity is wasted due to collisions, which is measured with 10 clients in their previous study, they use 0.85 as the maximum total CUE. Even if we assume the CUE is computed accurately, applying the fixed collision rate to CUETotalMax can result in critical problems because according to our measurement results in a test-bed, the collision rate changes from 5% to 60% even with the same number of VoIP sources. Also, it is difficult to correctly estimate the QoS of a new flow using the remaining CUE value.

Zhai et al. [30] proposed a call admission scheme using the channel busyness ratio (R_b), the ratio of the time that a channel is determined to be busy, which is similar with CUE. However, unlike CUE, R_b is computed in every client by looking at the actual MAC and PHY layer behavior. When a new call is requested, the transmission rate is changed to the average channel utilization (CU) and peak channel utilization (CU_{peak}) and they are sent to the AP. Then, the AP computes the total CU and CU_{peak} and compare it with the maximum CU, which was measured in advance. However, the maximum CU varies according to the traffic type and channel condition and the wrong maximum CU wastes bandwidth or impairs the QoS. Also, according to their simulation results, 10% of the resources were wasted after the admission control, which shows the inefficiency of the call admission control algorithm.

Kuo et al. [20] used an analytical model to decide the admission of a new call. When a new call is requested, the expected bandwidth and delay are computed using an analytical model. However, the assumption used in the analytical model has the same problem as [23].

2 Balancing uplink delay and downlink delay using APC

As mentioned in the introduction, when the channel gets congested, for VoIP traffic the downlink increases significantly while the uplink delay remains very low. Fig. 5 shows the uplink and downlink delay of CBR VoIP traffic measured in the ORBIT test-bed[2] with 15 VoIP sources, which the capacity for the VoIP traffic[3]. We can see that while the downlink delay rises above 300 ms in worst case, the uplink delay increases to at most 100 ms and it happens very rarely. Fig. 6 also shows that as the number of VoIP sources increases, the downlink delay increases significantly while the uplink delay remain very low.

[2] http://www.orbit-lab.org

[3] 64 kb/s voice bit rate and 20 ms packetization interval in 802.11b with the fixed 11 Mb/s transmission rate and the short preamble

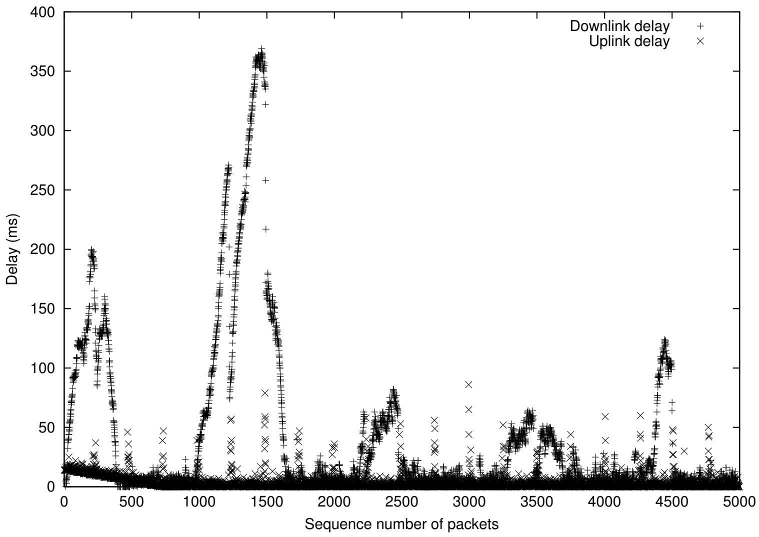


Figure 5: Uplink and downlink delay of 64 kb/s CBR VoIP traffic in DCF

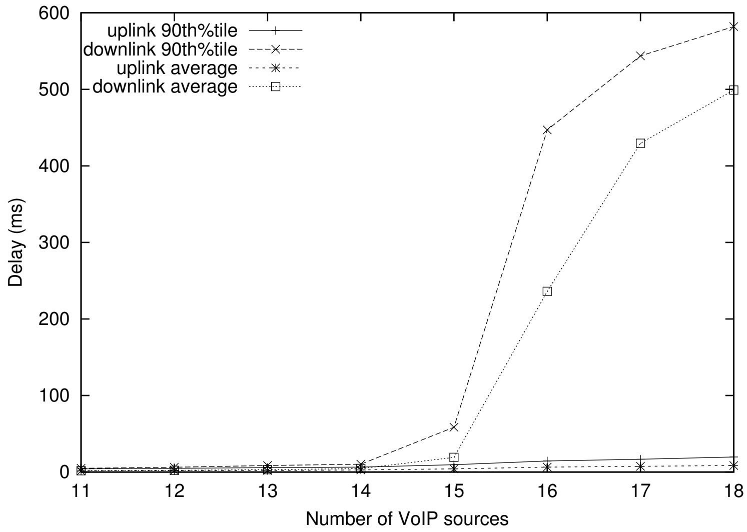


Figure 6: The delay of VoIP traffic in DCF

The imbalance between the uplink and downlink delay is caused by the unfair resource distribution between the uplink and downlink in DCF; while the AP needs to forward all packets to all wireless clients and has far more packets than the clients, it has the same chance to access media with clients under CSMA/CA. Because of this, when the channel is congested due to any interference or too many simultaneous media access, only the downlink delay increases significantly. Therefore, by allowing more bandwidth to the downlink traffic, we can balance the uplink and downlink delay, which leads to the improvement of the capacity for VoIP traffic.

2.1 Adaptive Priority Control (APC)

Before I mention how to decide the optimal priority of the AP to balance the uplink and downlink delay, I discuss how to apply the priority of the AP at the MAC layer. This is because the methods to apply the priority usually cause overhead and the overhead affects the priority decision algorithm.

In IEEE 802.11, there are three well-known methods to control the priority of wireless nodes. All three methods are used in IEEE 802.11e in order to differentiate the priorities of frames according to the Access Category (AC). The first method is to control the contention window (CW) size. When nodes have a smaller window size, the backoff time becomes smaller and the transmission rate becomes higher. This method increases the collision rate unfortunately as the window size decreases [21], and it is difficult to accurately control the priority since the backoff time is chosen randomly within the CW size. The second method is to change the Inter-Frame Spacing (IFS). A smaller IFS causes the backoff time to decrease faster and the transmission rate to increase naturally, and the node with the smaller IFS wins the channel when two nodes try to transmit frames at the same time. However, it has the same problem as the first one because the backoff time is still decided randomly. The last method is the Contention Free Transmission (CFT) where clients transmit multiple frames contention free (without backoff) when a node gets a chance to transmit a frame and control the number of frames sent contention free. APC uses CFT because it allows us to control the transmission rate precisely according to the priority without overhead.

For fairness between the downlink (the AP) and uplink (wireless clients) in a BSS, when uplink and downlink have the same amount of traffic, the AP and the wireless clients need to be able to send the same number of packets within a given interval. Then, intuitively, the AP needs to send N frames while N wireless clients transmit a frame each. I call this approach 'semi-adaptive method'. In VoIP traffic, when a single packetization interval is used for all VoIP traffic in a BSS, the uplink and downlink traffic volumes are symmetric, in general with a large number of VoIP sources, and thus we can apply the semi-adaptive method to balance the uplink and downlink delay. However, when more than one packetization interval is used for VoIP traffic in a BSS, the traffic volume of the uplink and downlink becomes asymmetric: even when the number of active wireless clients and Ethernet clients are the same, the number of packets from them depends on the packetization intervals of the active clients. For example, when ten Ethernet clients with 10 ms packetization interval and ten wireless clients with 20 ms packetization interval are talking with the same 64 kb/s voice bit rate, the volume of the downlink traffic from Ethernet clients is larger than the uplink traffic volume because of the overhead such as packet headers, even if the total voice data size is identical. In such a case, we need to consider the traffic volume of uplink and downlink in deciding the AP's priority. I propose to use the ratio of the number of packets in the queue of the AP and an average queue size of all wireless nodes as the priority of the AP (P) when the queue of wireless nodes is not empty, and the number of active wireless nodes when the queue is empty. That is, P

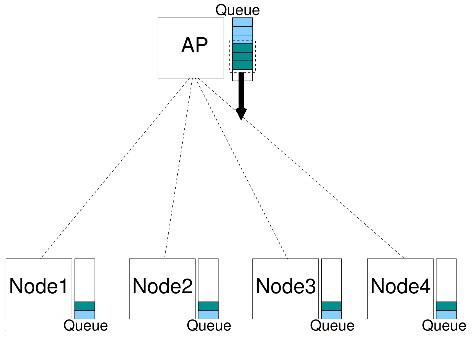


Figure 7: Packet transmission in APC

is calculated as follows:

$$P = \begin{cases} \lceil \frac{Q_{AP}}{Q_{Node}} \rceil & \text{if } Q_{Node} \geq 1 \\ N_e & \text{if } Q_{Node} = 0 \end{cases} \tag{1}$$

where, Q_{AP} is the queue size of the AP, Q_{Node} is the average queue size of the wireless nodes, and N_e is the number of active Ethernet nodes. Even though different encoding rates are used among VoIP sources, still this method works because the voice data takes only 8% 26% of the total frame size [18] depending on the encoding rate, and the effect of packet size difference becomes smaller when the overhead from retransmission is considered.

For instance (Fig. 7), if four wireless nodes, Node 1, Node 2, Node 3 and Node 4 have two packets in each queue, and the AP has six packets in the queue. Then, the average queue size of the wireless nodes is 2, and the priority of the AP becomes three (= 6/2). Thus, the AP sends three frames contention free when it gets a chance to transmit a frame. If we assume that every node got the same chance to transmit frame(s), then the average number of packets in the queue of the wireless nodes and the one of the AP becomes one and three, respectively, and both of them become zero after the next transmission. Therefore, transmitting Q_{AP}/Q_{Node} packets contention free results in the same packet processing time in the AP and wireless nodes, which means that the AP and wireless nodes have the same queuing delay. I will prove this theoretically for the more general case in the next section. Also, in this way, the priority of the AP changes adaptively when the traffic volume of the uplink and downlink changes. When the amount of traffic to the AP caused by Ethernet nodes increases, the queue size of the AP increases and the priority also increases to balance the downlink delay with the uplink delay. When the queue size of the nodes increases, the priority of the AP decreases.

2.2 Theoretical analysis

In this section, I show that the AP needs to transmit Q_{AP}/Q_{Node} packets when $Q_{Node} > 1$ and N_e packets when $Q_{Node} = 0$ to balance the uplink and downlink delay.

I define the symbols used in the analysis as follows:
ΔQ_{AP} = Change of the number of packets in the queue of the AP
Q_{AP} = Number of packets in the queue of the AP
Q_{Node} = Average number of packets in the queue of all wireless nodes

D_{AP} = Queuing delay of the AP
D_{Node} = Queuing delay of a node
N_e = Number of active (talking) wired nodes
x_{AP} = Transmission overhead (backoff, deferral and retry) at the AP
i = Packetization interval
t = Transmission time of one VoIP frame including ACK
λ = Packet arrival rate
μ = Packet transmission rate
P = Priority of the AP to balance the uplink and downlink delay

The dominant component of delay is the queuing delay considering that the transmission delay and the transmission overhead are very small. Furthermore, the transmission delay is the same in the AP and wireless nodes assuming that they use the same transmission rate, and the transmission overhead, which includes backoff, deferral and retransmission overhead, is also similar for the AP and wireless nodes, while the queuing delay of the AP is much bigger than the one of the wireless nodes. Therefore, balancing the queuing delay of the AP and wireless nodes results in the balanced uplink and downlink delay. Thus, I show that APC balances the queuing delay of the AP and wireless nodes.

We can compute the queuing delay by multiplying the transmission time to the queue size according to Little's law ($D_{system} = Q_{system}/\mu_{system}$). Without using the law, we can easily infer that the queuing delay can be computed by multiplying the queue size by the transmission rate. Then, we can compute the queuing delay of the AP (D_{AP}) and the nodes (D_{Node}) as follows:

$$D_{AP} = Q_{AP} \cdot \frac{1}{\mu_{AP}}$$

$$D_{Node} = Q_{Node} \cdot \frac{1}{\mu_{Node}}$$

We consider the priority of the AP (P) in two cases: When the queue size of nodes is greater than zero ($Q_{Nodes} \geq 1$) and when the queue size of nodes is zero ($Q_{Nodes} = 0$).

2.2.1 When $Q_{Nodes} \geq 1$

When all wireless nodes including the AP have packets to transmit, every wireless node as well as the AP has the same chance to transmit packets due to the fairness of CSMA/CA, that is, $\mu_{AP} = \mu_{Node}$, in DCF. Then, in APC, $\mu_{AP} = P \cdot \mu_{Node}$ because the AP transmits P packets when it gets a chance to transmit packets while each node transmits only one packet. Thus, D_{AP} can be rewritten as:

$$D_{AP} = Q_{AP} \cdot \frac{1}{P \cdot \mu_{Node}}$$

Then, we can get the optimal P value for balancing the delay of wireless nodes and the AP as follows:

$$D_{AP} = D_{Node}$$

$$Q_{AP} \cdot \frac{1}{P \cdot \mu_{Node}} = Q_{Node} \cdot \frac{1}{\mu_{Node}}$$

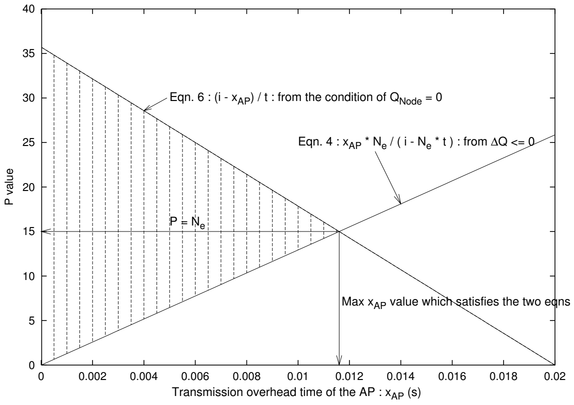


Figure 8: Optimal P value when Q_{Node} = 0

Then,

$$P = \frac{Q_{AP}}{Q_{Node}}$$

2.2.2 When $Q_{Node} = 0$

The queue size of wireless nodes decreases when the transmission rate of nodes at the MAC layer is bigger than the packet generation rate at the application layer, that is $\mu_{Node} \geq 1/i$, which is always satisfied if $Q_{Node} = 0$. In order to bring the queuing delay of the AP down to zero, the change of the queue size of the AP needs to be less than or equal to zero ($\Delta Q_{AP} \leq 0$). I derive the equation for ΔQ_{AP} to get the priority value of the AP (P) that satisfies it.

The change of the number of packets in the queue of the AP (ΔQ_{AP}) is the packet arrival rate to the AP minus the packet transmission rate from the AP:

$$\Delta Q_{AP} = \lambda_{AP} - \mu_{AP}$$

When the AP sends P packets contention free, the transmission time of a packet is $(x_{AP}+t\cdot P)/P$, and transmission rate (μ_{AP}) becomes $P/(x_{AP} + t \cdot P)$. Then, ΔQ_{AP} is rewritten as follows:

$$\Delta Q_{AP} = \frac{N_e}{i} - \frac{P}{x_{AP} + t \cdot P} \tag{2}$$

Here, for $\Delta Q_{AP} \leq 0$,

$$\frac{N_e}{i} \leq \frac{P}{x_{AP} + t \cdot P} \tag{3}$$

$$P \geq \frac{N_e \cdot x_{AP}}{i - N_e \cdot t} \tag{4}$$

According to Eqn. 4, P value is proportional to the transmission overhead of the AP as shown in Fig. 8: if the AP gets a chance to transmit packets very fast, the AP can transmit a small number of packets contention free, and if it takes a long time, it needs to transmit a large number of packets contention free.

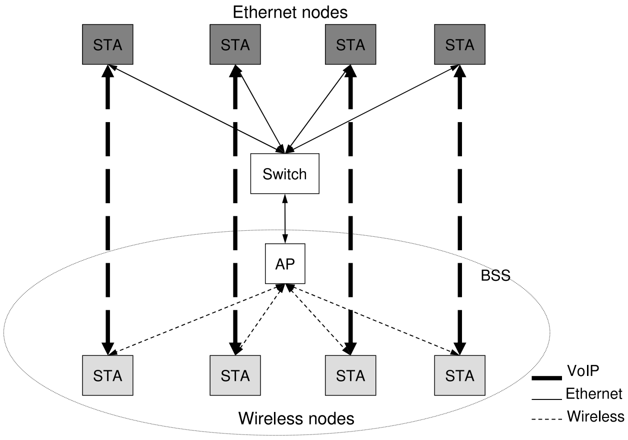


Figure 9: Simulation topology

Here, the transmission time of the AP should not exceed the packetization interval because the wireless nodes need to send at least a packet within a packetization interval to keep their queues empty. That is,

$$x_{AP} + t \cdot P < i \tag{5}$$

Then,

$$P < \frac{i - x_{AP}}{t} \tag{6}$$

Eqns 4 and 6 are plotted in Fig. 8 with $t = 0.00056\,\mathrm{s}^4$, $N_e = 15$ and $i = 0.02\,\mathrm{s}$, and the shaded region represents the one that satisfies Eqn. 4 and Eqn. 6. According to the two graphs in Fig. 8, we can see that P should be less than or equal to N_e. We can also get the same result when we combine Eqn. 2 and Eqn. 5:

$$\frac{N_e}{i} = \frac{P}{x_{AP} + t \cdot P} \geq \frac{P}{i}$$
$$P \leq N_e$$

Therefore, the optimal P value that satisfies the two equations for all possible values of x_{AP} is N_e.

2.3 Simulation and results

In order to evaluate the performance of APC, I have implemented APC in the QualNet simulator [24] and measured the uplink and downlink delay with various packetization intervals.

2.3.1 Simulation parameters

As shown in Fig. 9, I used the Ethernet-to-wireless network topology to focus on the delay in a BSS. In the simulations, the Ethernet portion added 1 ms of transmission delay, which allows us to to assume that the end-to-end delay is essentially the same as the wireless transmission delay.

[4] The t value is calculated with 160B (20 ms packetization interval and G.711 codec) payload in 11 Mb/s transmission rate

Table 1: Parameters in IEEE 802.11b (11 Mb/s)

Parameters	value
PLCP[5]Preamble	72.00 (short) μs
PLCP Header	48.00 μs
PLCP Header Service	192.40 μs
MAC Header+CRC	36 B
RTS threshold	1500 B
Retransmission limit	7 frame
SIFS	10 μs
DIFS	50 μs
Slot	20μs
CW_{MIN}	31 slots

Table 2: Voice pattern in ITU-T P.59 (Temporal parameters in conversational speech)

Parameter	Duration (s)	Fraction (%)
Talkspurt	1.004	38.53
Pause	1.587	61.47
Double Talk	0.228	6.59
Mutual Silence	0.508	22.48

I used IEEE 802.11b [11] and the parameters are shown in Table 1. VoIP packets are encoded using G.711 codec with payloads of 80, 160, and 320 bytes, which represent packetization intervals of 10, 20 and 40 ms, respectively (voice bit rate of 64 kb/s). I added an additional 12 bytes to the payload reflecting the overhead incurred by RTP [25] header. The VoIP packets were transported with UDP. I considered VoIP traffic with silence suppression, using the conversational speech model with double talk described in ITU-T P.59 [15]. The parameters are shown in Table 2 and the conversation model is shown in Fig. 10.

[5] Physical Layer Convergence Protocol

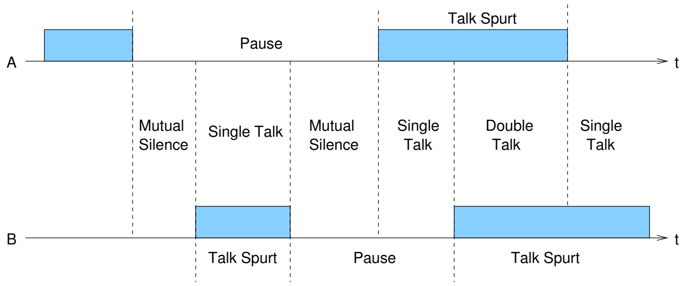
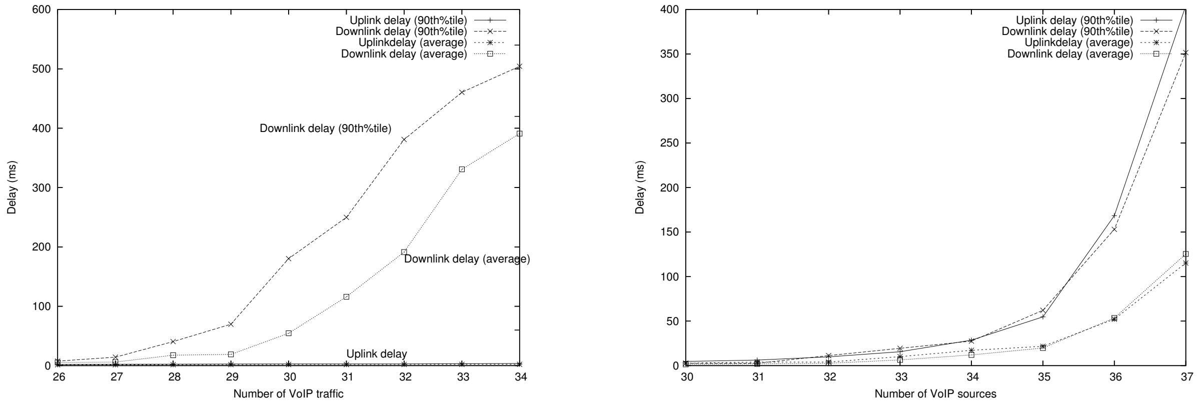


Figure 10: Conversational speech model in ITU-T P.59



(a) DCF

(b) APC

Figure 11: Simulation results of DCF and APC with 20 ms packetization interval for 64 kb/s VoIP traffic

2.3.2 Capacity for VoIP traffic

The one-way end-to-end delay of voice packets should be less than 150 ms [14] [27]. I assumed the codec delay to be about 30-40 ms at both the sender and the receiver, and the backbone network delay to be about 20 ms. Therefore, the wireless network should contribute less than about 60 ms to the total end-to-end delay.

I measured the 90th percentile value[6] of the uplink and downlink delay of voice packets, and defined the capacity of VoIP as the maximum number of wireless nodes so that the average of the 90th percentile of the one-way end-to-end delay for both direction does not exceed 60 ms.

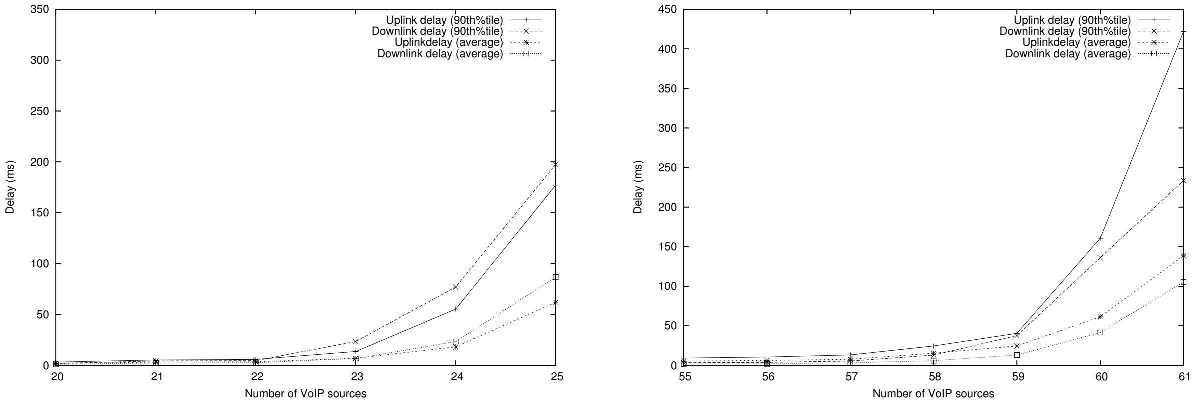
2.3.3 Simulation results

Fig. 11 shows the simulation results with 20 ms packetization interval and 64 kb/s VoIP traffic. I plotted both the 90th percentile and average value of uplink and downlink delay because the 90th percentile value is a good measure of the capacity for the VoIP traffic, and the average value is used to check the balance of the uplink and downlink delay. Fig. 11 shows that APC balances the uplink and downlink delay effectively. And if we compare with the result of DCF (Fig. 11(a)), we can see that APC improves not only the balance between uplink and downlink delay but also the capacity for the VoIP traffic by 25 %, from 28 calls to 35 calls.

Fig. 12 shows the simulation results using 10 ms and 20 ms packetization intervals on both Ethernet and wireless nodes half and half, and 40 ms packetization interval only. We can see that APC balances the uplink and downlink delay in both cases. This shows that APC changes the priority of the AP adaptively to the change of the uplink and downlink traffic volume.

In order to see how the uplink and downlink delays change with simulation time, I have plotted the two components throughout the simulation time, and I have confirmed that uplink and downlink

[6] Generally, 90th percentile value is used for measuring QoS of VoIP because it indicates the jitter of VoIP applications.



(a) 20 ms and 40 ms packetization intervals

(b) 40 ms packetization interval only

Figure 12: Simulation results for APC with various packetization intervals

delay are balanced throughout the whole simulation. Fig. 13 shows a sample simulation result with 36 VoIP sources (64 kb/s and 20 ms packetization interval).

2.4 Conclusion

I have shown that as the number of VoIP sources increases, the downlink delay increases significantly while the uplink delay remains low in DCF. This is because every wireless node including the AP has the same chance to transmit frames in DCF, while the AP needs to transmit more packets than wireless nodes. In this proposal, I have proposed APC, which differentiates the priority of the AP from the wireless nodes adaptively according to the traffic volume and balances the uplink and downlink delay, by allowing the AP to transmit Q_{AP}/Q_{Node} packets contention free. I have also analyzed the performance of APC theoretically and have proved that APC balances the uplink and downlink delay.

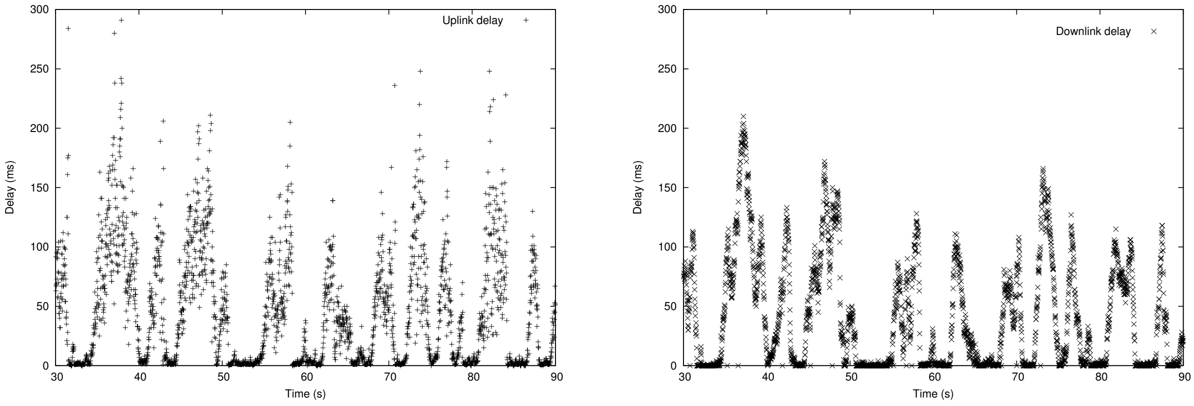
I have implemented the APC algorithm using the QualNet simulator and have shown that APC balances the uplink and downlink delay effectively in VoIP traffic with various packetization intervals.

2.5 Future work

I have shown that APC works very well with only VoIP flows. However, in real environments, VoIP traffic and other background traffic such as email, HTTP and P2P coexist. In such situations, IEEE 802.11e needs to be used to prioritize the traffic according to traffic types. I will implement the APC under 802.11e and evaluate the performance with various types of background traffic.

Also, I will implement the APC in actual wireless clients using the MadWifi wireless card driver, and evaluate the performance in the ORBIT testbed and compare the results with the simulation results.

The APC requires knowledge of the queue size of the wireless clients in real time. I will investigate a method where the AP can achieve the same performance without using the actual



(a) Uplink delay

(b) Downlink delay

Figure 13: The uplink and downlink delay (90th%tile) with 36 VoIP sources using APC

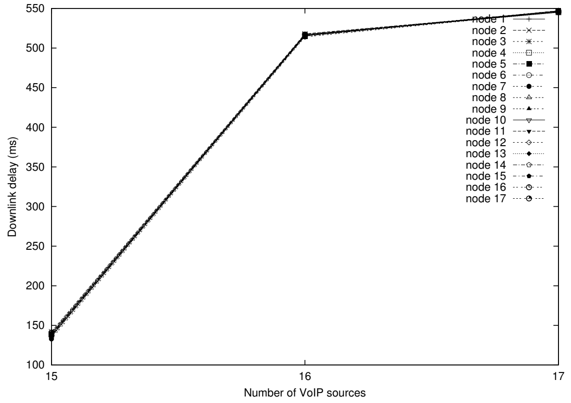


Figure 14: The downlink delay of all VoIP flows

queue size of the clients.

3 Call admission control with QP-CAT

When the number of flows reaches the capacity of the channel, the QoS of all VoIP flows can be significantly degraded by a newly admitted flow. Fig. 14 shows the downlink delay of all VoIP flows with 15 to 17 VoIP sources[7]. We can see that when a new VoIP flow is added to 15 VoIP sources, which is the capacity for VoIP traffic, the delay of all the VoIP flows drastically increases. This is because the increased transmission time due to collisions and the binary exponential increase of the backoff time causes a significant increase of the queuing delay of the downlink packets. This shows that a call admission control mechanism is necessary to protect the QoS of existing VoIP flows. Call admission control in IEEE 802.11 is totally different from that in Ethernet due to the characteristics of CSMA/CA. While most of the admission control algorithms for wired networks

[7] 64 kb/s CBR VoIP traffic with 20 ms packetization interval in 802.11b under 11 Mb/s transmission rate

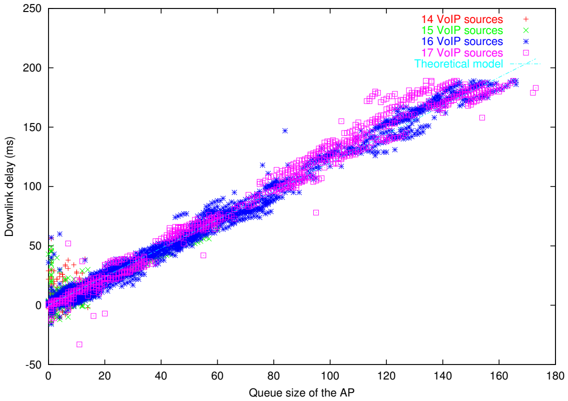


Figure 15: Correlation between the queue size of the AP and instant downlink delay

are based on the end-to-end QoS, that in wireless networks is mainly to protect the QoS of flows between the AP and clients in a BSS. Therefore, most of the legacy admission control algorithms for wired networks are not applicable. Further, the admission decision in wireless networks is more difficult than that in wired networks. For the admission decision, the achievable throughput with a limited amount of delay needs to be estimated, but it is hard to predict it in wireless networks because it changes according to the overhead incurred by collisions as well as the number of active nodes and the parameters of VoIP flows. Therefore, an accurate metric to estimate the channel condition is essential for admission control in wireless networks.

3.1 Correlation between the queue size and downlink delay

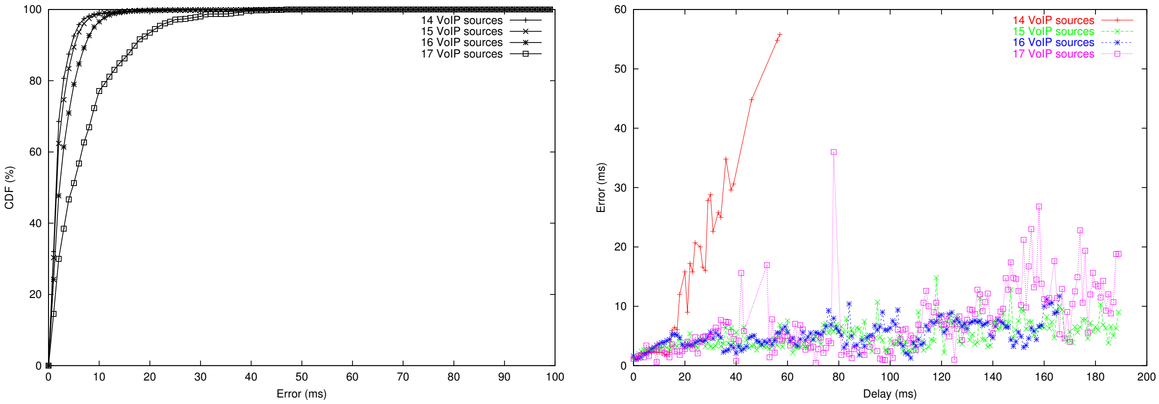
An accurate metric to estimate the channel condition is the most critical factor for the call admission control, and our approach uses as the metric the queue size of the AP, which is easy to compute and allows the most accurate estimation of the QoS of all VoIP flows. To verify it, I identified the correlation between the queue size of the AP and the downlink delay of VoIP traffic.

In order to identify the correlation, I measured the queue size of the AP and downlink delay in the ORBIT test-bed. I used 64 kb/s CBR VoIP traffic with 20 ms packetization interval and the fixed 11 Mb/s transmission rate in 802.11b. All the 802.11 parameters used in the experiments are summarized in Table 1. Also, 14 to 17 VoIP sources were used in the measurement because the channel starts to be congested at 14 or more VoIP sources. Fig. 15 shows the experimental results. We can see that the queue size of the AP and downlink delay are strongly correlated each other; as the queue size of the AP increases, the downlink delay also linearly increases.

I have verified the correlation also by numerical analysis. The downlink delay (D) is composed of the queuing delay (D_Q), transmission delay (D_T) and propagation delay. We can ignore the the propagation delay because it is very small. Then, $D = D_Q + D_T$.

We can compute the queuing delay (D_Q) by multiplying the transmission delay (D_T) to the queue size (Q) from Little's law ($D_{system} = Q_{system}/\mu_{system}$). Then, we can compute the queuing delay of the AP (D_{AP}) and the nodes (D_{Node}) as follows:

$$D_Q = Q \cdot \frac{1}{\mu_{AP}} = Q \cdot D_T$$



(a) CDF

(b) Error according to delay

Figure 16: The errors between the actual downlink delay and the estimated one using the queue size of the AP

Therefore, the downlink delay (D) becomes

$$D = Q \cdot D_T + D_T = (Q + 1) \cdot D_T,$$

where D_T is the time to transmit one VoIP packet including all overhead. Even though the overhead slightly changes according to the channel condition, we can use an average value because the change is minor in VoIP traffic unless the channel is extremely overloaded. In Fig. 15, the straight line is the theoretical relationship between the queue size of the AP and downlink delay of the VoIP traffic. We can see that the experimental results are exactly following the theoretical model.

In order to identify the accuracy of the model, the cumulative distributed function of the errors between the actual downlink delay and the estimated one using the theoretical model was computed and plotted in Fig. 16(a). We can see that the 95%tile error is below 10 ms with 14 to 16 VoIP calls. With 17 VoIP sources, the error becomes larger because the channel is extremely overloaded and the transmission delay (D_T) increases significantly due to the increased overhead. However, this is not a problem because when the call admission control is applied appropriately, it never happens.

Fig. 16(b) shows the accuracy of the model according to the downlink delay. It shows that with 15 and 16 VoIP sources, the model keeps the similar accuracy even when downlink delay increases. With 14 VoIP sources, the errors for some packets increase above 50 ms, but this is acceptable because the frequency is negligible according to Fig. 16(a).

As the Figs 15 - 16(b) show, the downlink delay can be accurately estimated using the queue size of the AP and the theoretical model, and the queue size of the AP can be an accurate metric for the call admission control.

3.2 Queue size Prediction (QP) using Computation of Additional Transmission (CAT)

The best way to decide the admission of a new VoIP flow is to see the queue size of the AP after it is admitted. However, it is not appropriate to disconnect the admitted flow when it was discovered

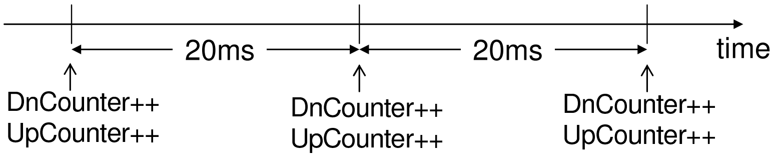


Figure 17: Emulation of a new VoIP flow with 20 ms packetization interval

that it deteriorates the QoS of all VoIP flows. Another way is that as in call admission control methods for wired networks, a probing flow can be transmitted instead of an actual VoIP flow, but it wastes a certain amount of bandwidth because it should keep probing in wireless networks, which is a critical disadvantage because of the limited bandwidth. Therefore, I propose a simple and accurate technique to predict the queue size of the AP by monitoring the channel, before a new VoIP flows is admitted.

The basic concept of the Queue size Prediction is to predict the future queue size of the AP using the emulation of a new VoIP flow and the Computation of Additional Transmission (CAT), where the number of packets additionally transmitted is computed by monitoring the channel status.

3.2.1 Emulation of new VoIP flows

In order to emulate a new VoIP flow, two counters, *UpCounter* and *DnCounter*, which count the number of the uplink and downlink packets of a new VoIP flow, respectively, are introduced. The counters are incremented following the behavior of actual VoIP flows. For example, for the VoIP traffic with 20 ms packetization interval, both of the counters are incremented by one every 20 ms (Fig 17). The counters are decremented in real time according to the number of packets computed using CAT. The two counters are decremented alternatively because the chance to transmit packets is the same between the uplink and downlink. Consequently, the actual queue size of the AP plus *DnCounter* becomes the predicted future queue size of the case when the VoIP flow is admitted.

3.2.2 Computation of Additional Transmission (CAT)

The number of additionally transmittable packets (n_p) is computed by looking at the current packet transmission behavior. A clock starts when medium becomes idle and stops when the busy medium is detected. When the clock stops, n_p is computed by dividing the clock time (T_c) by the total transmission time (T_t) of a VoIP packet (Eqn. 7) and deducted from the two counters. The transmission of a VoIP packet entails all headers in each layer, IFSs, backoff and an ACK frame. Thus, the transmission time (T_t) is computed as follows:

$$T_t = T_{DIFS} + T_b + T_v + T_{SIFS} + T_{ACK},$$

where T_v and T_{ACK} are the time for sending a voice packet and an ACK frame, respectively, T_b is the backoff time, T_{DIFS} and T_{SIFS} are the durations of DIFS and SIFS, respectively. The backoff time is *Number of Backoff Slots* $\times T_{Slot}$ where T_{slot} is a slot time, and *Number of Backoff Slots* has a uniform distribution over $(0, CW_{MIN})$ with an average of $(T_{Slot} \times CW_{MIN}/2)$ (Fig. 18).

$$n_p = \lfloor T_c/T_t \rfloor \tag{7}$$

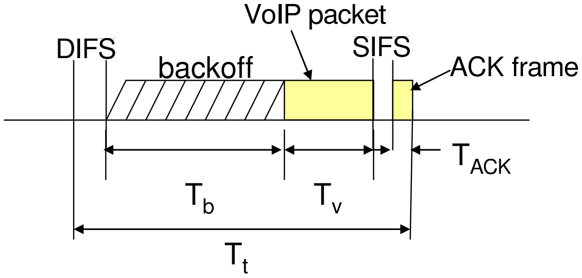


Figure 18: The transmission time of a VoIP packet

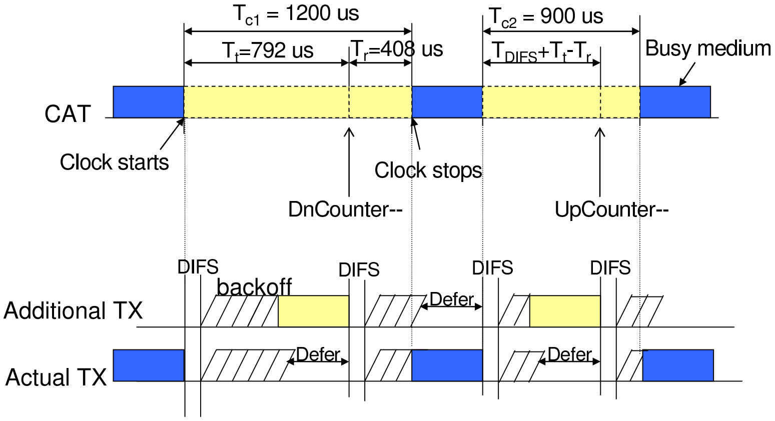


Figure 19: Computing the number of additionally transmittable VoIP packets

For example, with 64 kb/s VoIP traffic with 20 ms packetization interval, the voice data size is 160 B, the VoIP packet size including IP, UDP and RTP headers becomes 200 B, and then the total transmission time becomes 791.82 μs including the average backoff time (310 μs) and 14 B ACK frame (130.18 μs) in IEEE 802.11b with 11 Mb/s transmission rate (Refer to Table 1 for the 802.11b parameters). Thus, for example, if T_c is 1200 μs, then n_p is 1 according to Eqn. 7.

3.2.3 Emulation of deferral

In Fig. 19, when $T_c = 1200\mu s$, one additional packet can be transmitted ($n_p = 1$) and 408 μs still remains. The surplus time is computed as Eqn. 8 and accumulated to the next clock time.

$$T_r = T_c - n_p \times T_t \tag{8}$$

For example, in Fig 19, the surplus time 408 μs is added to the next clock time 900 μs. In the computation of the next n_p, another T_{DIFS} is added to T_t. This is to emulate the deferral of the transmission. If we see the second figure in Fig. 19, we can notice that the computation method is consistent with the actual transmission behavior. During T_r, the backoff timer is decremented, and when busy medium is detected, the transmission needs to be deferred. When the media becomes idle again, the backoff time is decremented after DIFS, which is the second DIFS in the transmission. We can see that as in the computation of n_p, two packets can be additionally transmitted during the idle time in the actual transmission.

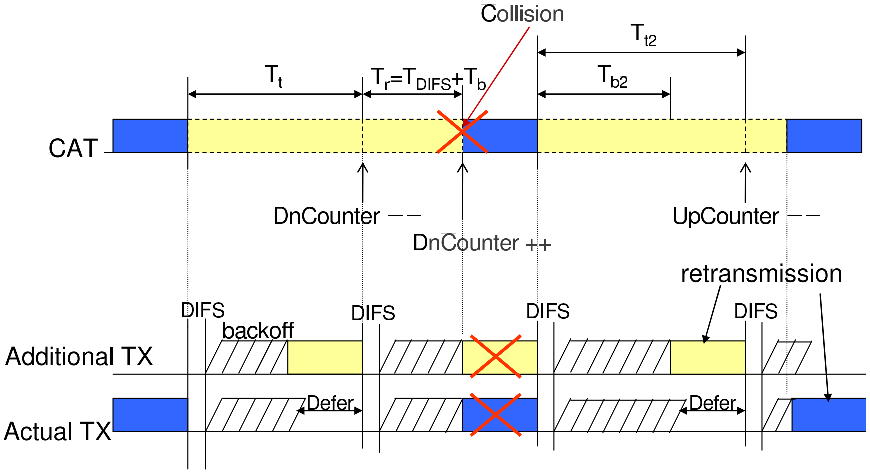


Figure 20: Emulation of collisions

3.2.4 Emulation of collisions

To predict the queue size of the AP more accurately, we need to consider collisions, which make channels more congested and cause additional delay. A simple way to emulate collisions is to use the current average collision rate for a certain amount of time. For example, if the average collision rate of downlink traffic is 3%, then three is added to the counters every time the counters are incremented by 100. However, while this method is easy to implement, it cannot reflect the increase of collisions due to the admission of a new call. Therefore, in CAT we eumulate collisions following the actual collision mechanism.

As we can see in Fig. 20, if T_r is exactly same with the backoff time and DIFS (that is, $T_r = T_b + T_{DIFS}$), then it is considered as a collision, because when a VoIP packet is transmitted at that point (right after T_b), an actual collision occurs. When a collision is detected, the backoff time is computed with the increased contention window size again, and the counters are not decremented. Additionally, to emulate the retransmission of the collided packet of existing flows, the downlink packet counter (*DnCounter*) is incremented by one, which emulates the delay of downlink packet due to the retransmission of the collided packet. The second figure in Fig. 20 shows the actual transmission behavior in case of collisions, which is consistent with the computation method.

3.2.5 Serialization of downlink packets

The last thing we need to consider is the serialization of downlink VoIP packets. The backoff time of an uplink packet and a downlink packet can be decremented at the same time, but those of two downlink packets cannot be decremented at the same time because they need to be transmitted one by one. Therefore, if the clock stops because of the transmission of a downlink packet and the *DnCounter* was decremented previously, then additional backoff time needs to be deducted from the remaining time. Fig. 21 shows the example. In the second packet transmission, the total transmission time becomes $T_t + T_{DIFS} + T_b$, that is, another backoff time was added. The second figure shows the actual transmission of two downlink packets, which takes $2 \cdot T_t$ as in the computation.

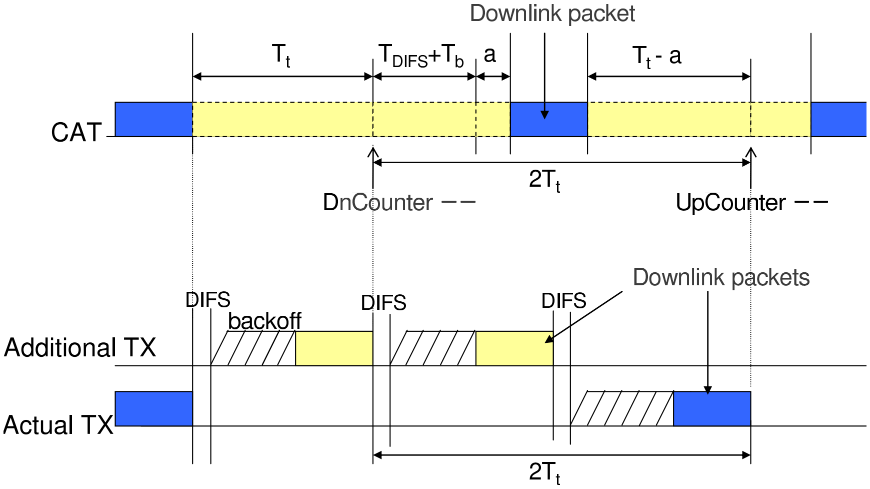


Figure 21: Serialization of downlink packets

3.3 Simulation results

I have implemented the QP-CAT using the QualNet 3.9 simulator[24], and evaluated the performance. The network topology in Fig. 9, 64 kb/s CBR VoIP traffic with 20 ms packetization interval, and the 802.11b parameters in Table 2 were used for simulations.

Fig. 22 shows the actual queue size of the AP with 15 VoIP calls, the predicted queue size of 16 VoIP sources, and the actual queue size with 16 VoIP sources, during 100 second simulation time. We can see that we can accurately predict the queue size of the AP when a new VoIP call is added to 15 VoIP calls.

I ran more simulations with various types of VoIP traffic and different number of VoIP sources and computed the average of both the actual and predicted queue size of the AP. As we can see in Fig 23, the CAT can predict the increase of the queue size of the AP in all the types of VoIP traffic when the number of VoIP sources exceeds the capacity of each VoIP traffic type.

However, this is an easy case where the delay is very small when the number of VoIP sources is below the capacity, and an additional VoIP source causes a significant increase of delay. In real environments where there are more interferences from walls or people, for example, even with 15 calls with 64 kb/s 20 ms packetization interval VoIP traffic (Fig 23(a)), the delay significantly fluctuates according to the link condition of each flow. Thus, an additional call to 14 VoIP calls can degrade the overall QoS of all flows according to the link condition and the 15th call needs to be rejected in the case.

In order to check the performance of CAT in such situations, I changed the transmission interval of VoIP packets among clients to cause more collisions. Fig 24 shows the simulation results with 64 kb/s 20 ms packetization interval VoIP traffic . Generally, as the transmission intervals become shorter, the more collisions occur and the delay increases. For example, when two wireless clients generate VoIP packets at the same time every 20 ms in the application layer, the chance that the frames collide each other is very high, but on the other hand, the packet generation interval between the two nodes is more than 620 ms (average backoff slots 31 × one slot time 20 ms), the two packets hardly collide each other. We can see that when the intervals decreases below 350 ms, the delay exceeds the threshold value for the capacity (the straight line), and the CAT can also predict the

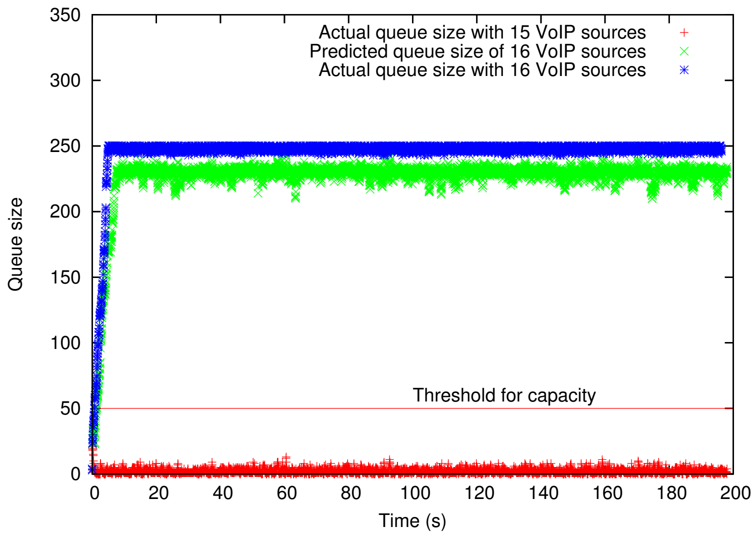


Figure 22: Simulation results of QP-CAT with 15 VoIP sources

increase very well. The QP-CAT a little bit overestimates the delay in 400 ms transmission interval, but this is not a problem because a little bit of conservativeness helps in protecting the existing VoIP flows.

3.4 Framework for call admission control

3.4.1 Admission control with IEEE 802.11e

IEEE 802.11e standard [13] defines a framework to acquire the admission of a new flow from QAP[8] (Fig. 25). Wireless clients needs to request the bandwidth required to transmit the flow through the ADDTS request. It contains the TSPEC that represents the traffic specification such as the minimum and maximum data size, the service rate and the data rate. Then, the QAP decides the admission using an admission control algorithm, which is not defined in the standard, and transmits the ADDTS response with the allowed TSPEC.

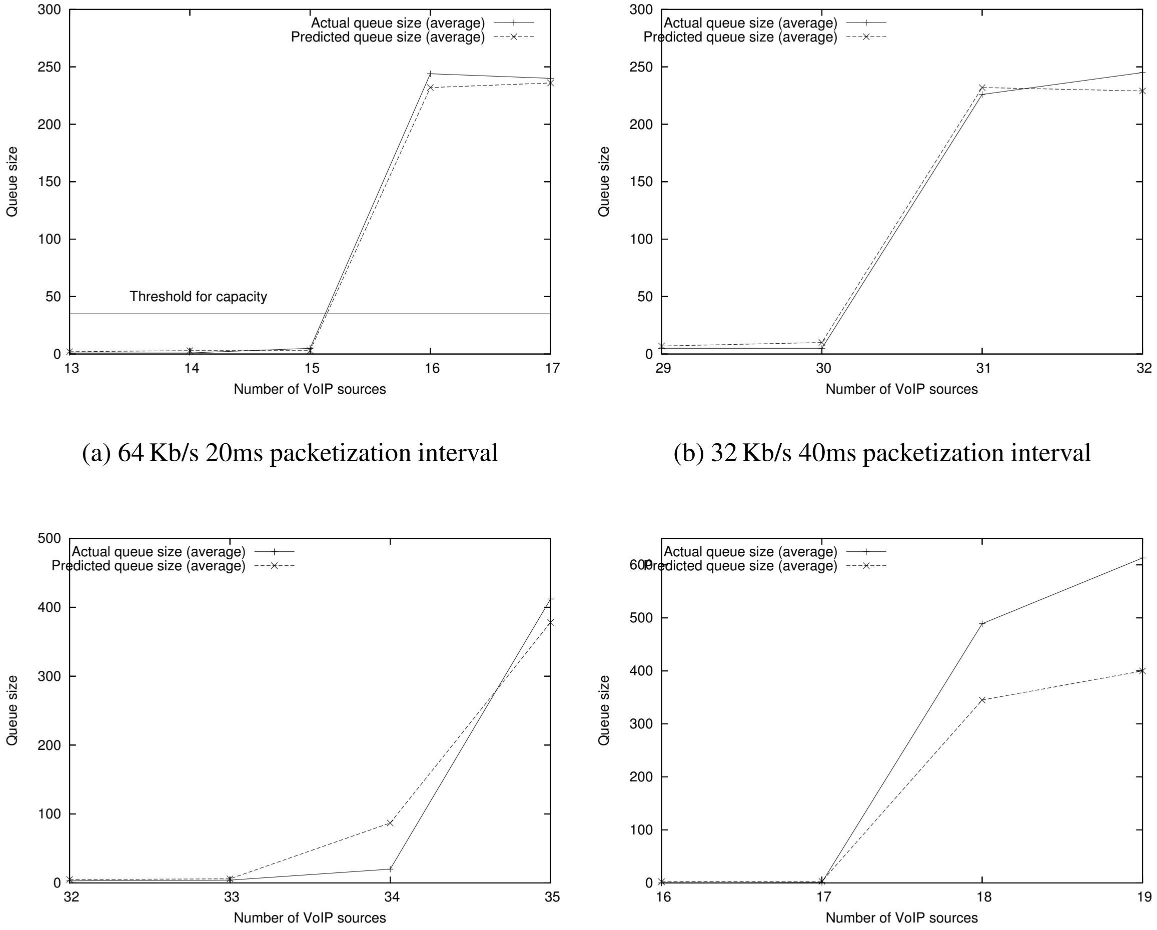
However, the procedure is not enough for the call admission control. Fig. 26 shows the problems. In both of the scenarios, wireless clients find that the requested resources are not available after a call set is established. Then, the call is dropped after the callee responded the call, which is not desirable. We can solve the problems using SIP precondition (RFC 3312) [5].

3.4.2 SIP precondition

SIP precondition (RFC 3112) was proposed to reserve the resources between the caller and callee before making a call to avoid the above problems. As we can see in Fig 27, the caller sends INVITE message with the desired bandwidth information (SDP1[9]). When the callee agrees to reserve the requested bandwidth, the callee sends the Session Progress message to the caller asking to reserve the resources including the desired resource information (SDP2). When the caller successfully

[8] The AP that supports 802.11e standard

[9] Session Description Protocol (SDP)



(a) 64 Kb/s 20ms packetization interval

(b) 32 Kb/s 40ms packetization interval

(c) 16 Kb/s 40ms packetization interval

(d) 16 Kb/s 20ms packetization interval

Figure 23: Simulation results of QP-CAT with various types of VoIP traffic

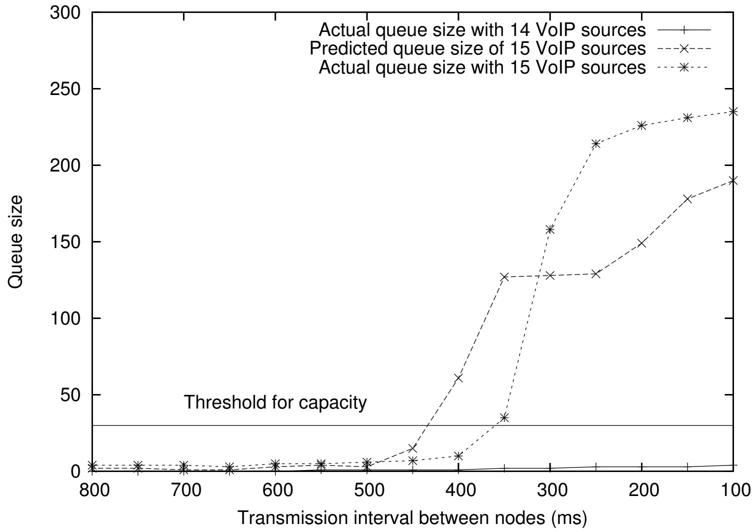


Figure 24: Simulation results of QP-CAT with 14 VoIP sources with various transmission intervals

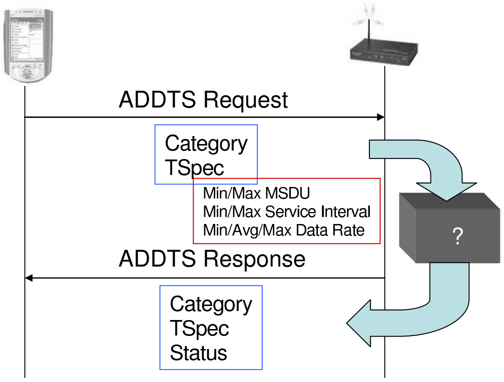
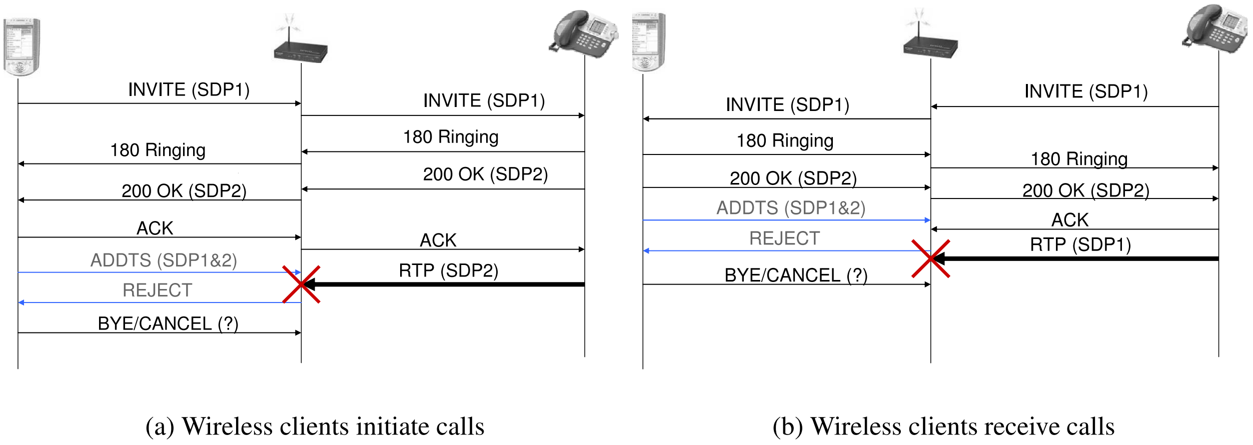


Figure 25: The framework of call admission control in 802.11e



(a) Wireless clients initiate calls

(b) Wireless clients receive calls

Figure 26: The problems of the admission control framework in IEEE 802.11e

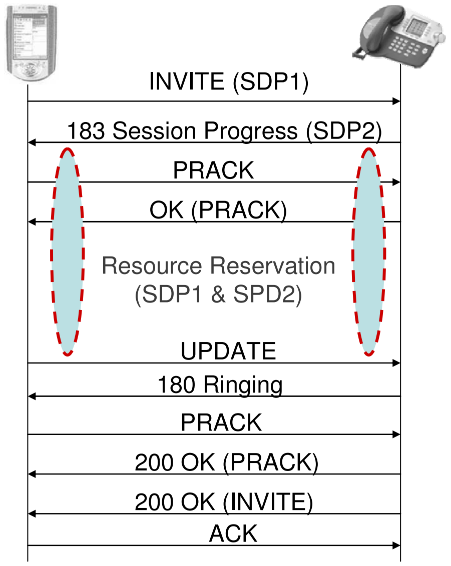


Figure 27: The framework of SIP precondition (RFC 3112)

reserved the requested resources, it sends the SIP UPDATE message and the callee responds with the OK message indicating that all the preconditions for the session have been met. At this point, the callee can alert the user and establishes the session in a normal way.

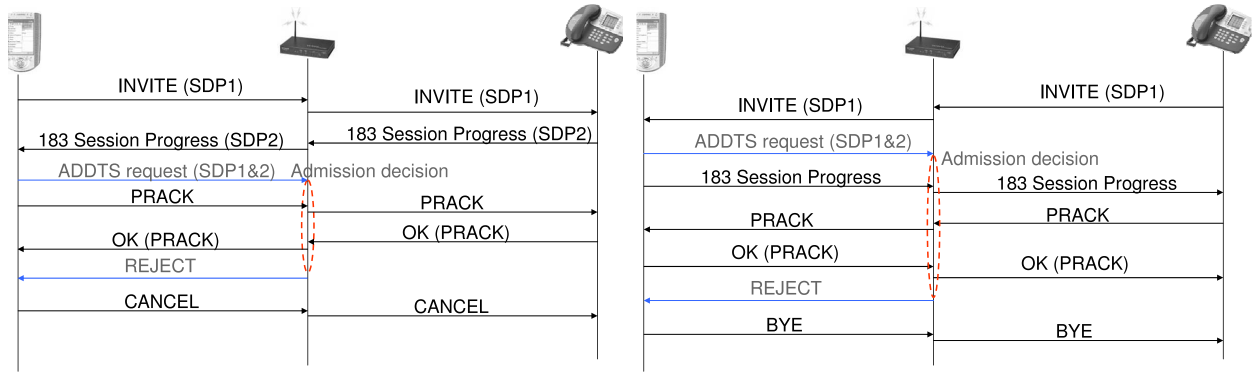
We can solve the problems mentioned in the last section by combining the SIP precondition and the 802.11e access control mechanism. Fig. 28 shows the procedures to deny calls without users being alerted using the SIP precondition. Wireless clients request resources for both uplink and downlink after getting the session information of the callee, and the callee does not notify the call before the requested resources are granted.

However, we can notice that many messages need to be exchanged even when the call is dropped. The messages not only cause the long delay, but also waste the bandwidth in wireless networks. Therefore, we need to reduce the messages and delay if possible.

The ideal solution is to implement the SIP stack at the AP so that it can recognize the requested resources from the INVITE message and make admission decisions without forwarding any SIP messages to the callee (Fig. 29(a)). Another practical solution is to use the SIP proxy server. When the SIP proxy server receives the INVITE message from a caller, it finds the AP the callee is associated with from a presence table and sends to the AP the ADDTS request with both uplink and downlink resource information. This method eliminates the message exchange between the AP and wireless clients and avoid wasting bandwidth of wireless networks. However, the AP needs to be changed to accept the ADDTS request from the SIP proxy server.

3.5 Conclusion

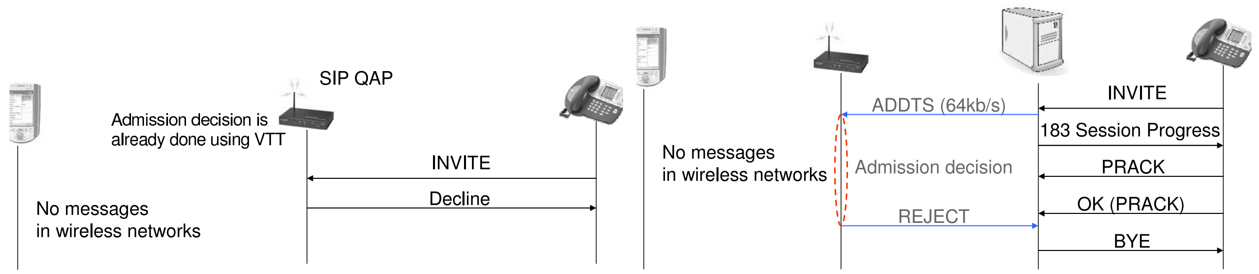
I have proposed an efficient call admission control using the queue size prediction technique. It uses as the metric the queue size of the AP, which has a strong correlation with the downlink delay. It predicts the queue size of the AP when a new VoIP flow were admitted, by computing the number of packets to be additionally transmitted. It has the similar performance with the actual probing, which generally gives the best estimation of the QoS, without any waste of bandwidth. Further, it



(a) Wireless clients initiate the call

(b) Wireless clients receive the call

Figure 28: The call admission control together with SIP precondition and 802.11e



(a) Wireless clients initiate the call

(b) Wireless clients receive the call

Figure 29: The call admission control together with SIP precondition and 802.11e

can simultaneously estimate the effect of multiple types of VoIP traffic by running multiple QP-CATs in parallel, which cannot be achieved in the actual probing approaches.

In order to evaluate the performance, I implemented the QP-CAT in the QualNet simulator 3.9, and run simulations with a different number of VoIP sources in various channel conditions. I have shown that it can accurately predict the effect of the additional VoIP flow on the existing flows so that the AP can make correct admission decisions.

3.6 Future work

The QP-CAT was evaluated with various number of VoIP sources and conditions, but only VoIP traffic was used in the simulations. Because in real environments, usually VoIP traffic and background traffic coexist, I will evaluate it with various types of background traffic.

In order to make the best admission decision, another step is required in addition to the accurate prediction of the queue size of the AP. When the number of VoIP sources is far from the capacity, it is easy to make a decision, but as it reaches the capacity, the queue size fluctuates around the border of the threshold and it takes time to make accurate decisions. Therefore, I will introduce an accurate and fast decision making system and evaluate the efficiency using the actual VoIP call arrival and duration patterns, which is missing in most of the previous related papers.

Finally, I would like to implement the QP-CAT in actual wireless clients, and evaluate the performance in the ORBIT testbed and compare the results with the simulation results.

4 Research plan

Regarding the APC, I finished the implementation of the current version of the APC in QualNet simulator 3.9 and evaluated the performance with various types of VoIP traffic. However, I still need to evaluate the APC under various background traffic and improve it, if needed. Also, I will implement the APC using the MadWifi driver and evaluate it in the ORBIT testbed.

In terms of the QP-CAT, I finished the implementation in the QualNet simulator and evaluated using a type of VoIP traffic. I will evaluate it with various types of VoIP traffic together with background traffic. And, I will implement the call admission control framework using QP-CAT and evaluate the efficiency by emulating the actual VoIP call arrival patterns and duration time. Additionally, I would like to implement it with actual wireless clients and evaluate it in the ORBIT-test bed.

4.1 Plan for completion of the research

Table 3 shows my plan for completion of the research.

Thus, I plan to defend my thesis in February 2008.

Timeline	Work	Progress
	Implement the QP-CAT in the QualNet simulator	completed
	Implement the APC in QualNet simulator and evaluated the performance with various types of VoIP traffic	completed
Dec. 2006	Evaluate the performance of the QP-CAT using simulations with various types of VoIP traffic	ongoing
Jan. 2007	Apply QP-CAT to call admission control and evaluate the efficiency	
Mar. 2007	Evaluate the call admission control using QP-CAT with various types of background traffic	
Apr. 2007	Integrate APC with 802.11e using the QualNet simulator	
May. 2007	Evaluate APC with various types of background traffic	
Jun. 2007	Implement the APC in the MadWifi wireless card driver	
Jul. 2007	Evaluate the performance of the APC in the ORBIT test-bed and compare it with the simulation results	
Aug. 2007	Introduce and implement the APC that does not use the queue size of clients	
Oct. 2007	Start to write my thesis	
Dec. 2007	Finish to write my thesis	
Feb. 2008	Defend my thesis	

Table 3: Plan for completion of my research

References

[1] W. Arbaugh A. Mishra, M. Shin. An Empirical Analysis of the IEEE 802.11 MAC Layer Handoff Process. *ACM SIGCOMM Computer Communication Review*, 33(2):93–102, April 2003.

[2] I Aad and C Castelluccia. Differentiation mechanism for IEEE 802.11. In *IEEE INFOCOM*, pages 209–218, Apr 2001.

[3] N. Akhtar, M. Georgiades, C. Politis, and R. Tafazolli. SIP-based end system mobility solution for all-IP infrastructures. In *IST Mobile & Wireless Communications Summit 2003*, June 2003.

[4] M Barry, A T Campbell, and A Veres. Distributed control algorithms for service differentiation in wireless packet networks. In *IEEE INFOCOM*, pages 582–590, Apr 2001.

[5] G. Camarillo, W. Marshall, and J. Rosenberg. Integration of Resource Management and Session Initiation Protocol (SIP). RFC 3312, IETF, Oct 2002.

[6] Casetti, C. Chiasserini, and C.-F. Improving fairness and throughput for voice traffic in 802.11e EDCA. In *Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004. 15th IEEE International Symposium*, volume 1, pages 525–530, 2004.

[7] D J Deng, R S Chang, and A Veres. A priority scheme for IEEE 802.11 DCF access method. *IEICE Trans. Commun.*, E82-B(1):96 – 102, Oct 1999.

[8] R. E. Droms. Dynamic Host Configuration Protocol (DHCP). RFC 2131, Internet Engineering Task Force, March 1997.

[9] Andrea Forte, Sangho Shin, and Henning Schulzrinne. Improving layer 3 handoff delay in IEEE 802.11 wireless networks. In *WICON 2006*, Aug 2006.

[10] Sachin Garg and M. Kappes. Admission control for VoIP traffic in IEEE 802.11 networks. In *GLOBECOM*, pages 3514–3518, Dec 2003.

[11] IEEE. *Wireless LAN Medium Access Control (MAC) and Physical (PHY) specifications*, 1999.

[12] IEEE. *Wireless LAN Medium Access Control (MAC) and Physical (PHY) specifications:Higher-Speed Physical Layer Extention in the 2.4GHz Band*, 1999.

[13] IEEE. *IEEE Draft Std. 802.11e, Medium Access Control (MAC) Enhancements for Quality of Service (QoS)*, D8.0 edition, Feb 2004.

[14] ITU-T G.114. *One-way Transmission Time*, 2003.

[15] ITU-T P.59. *Artificial Conversational Speech*, 1993.

[16] R Jain, D Chiu, and W Hawe. A quantative measure of fairness and discrimination for resource allocation in shared computer systems. Technical Report TR-301, DEC, 1984.

[17] Jiwoong Jeong, Sunghyun Choi, and Chong kwon Kim. Achieving weighted fairness between uplink and downlink in IEEE 802.11 DCF-based WLANs. In *Qshine*, August 2005.

[18] Takehiro Kawata, Sangho Shin, and Andrea G. Forte. Using dynamic PCF to improve the capacity for VoIP traffic in IEEE 802.11 networks. In *Wireless Communications and Networking Conference, 2005 IEEE*, volume 3, pages 13–17, Mar 2005.

[19] Sung Won Kim, Byung-Seo Kim, and Yuguang Fang. Downlink and uplink resource allocation in IEEE 802.11 wireless LANs. *IEEE Trans. on Vehicular Technology*, 54(1):320–327, Jan. 2005.

[20] Yu-Liang Kuo, Chi-Hung Lu, E.H.K. Wu, and Gen-Huey Chen. An admission control strategy for differentiated services in IEEE 802.11. In *GLOBECOM*, pages 707–712, Dec 2003.

[21] O.Tickoo and B.Sikdar. Queueing analysis and delay mitigation in IEEE 802.11 random access MAC based wireless networks. In *INFOCOM*, March 2004.

[22] S Pilosof, R Ramjee, D Raz, Y Shavitt, and P Sinha. Understanding TCP fairness over wireless LAN. In *IEEE INFOCOM*, pages 863–872, Mar 2003.

[23] Dennis Pong and Tim Moors. Call admission control for IEEE 802.11 contention access mechanism. In *GLOBECOM*, pages 174–178, Dec 2003.

[24] QualNet Network Simulator.

[25] Henning Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. Rtp: A transport protocol for real-time applications. RFC 3550, IETF, Jul 2003.

[26] Sangho Shin, Andrea G. Forte, Anshuman Singh Rawat, and Henning Schulzrinne. Reducing MAC layer handoff latency in IEEE 802.11 wireless LANs. In *MobiWac '04: Proceedings of the second international workshop on Mobility management & wireless access protocols*, pages 19–26, New York, NY, USA, 2004. ACM Press.

[27] TIA. *Voice Quality Recommendations for IP Telephony*, 2003.

[28] Xin Gang Wang, Geyong Min, Mellor, and J.E. Improving VoIP application's performance over WLAN using a new distributed fair MAC scheme. In *Advanced Information Networking and Applications, 2004. AINA 2004. 18th International Conference*, volume 1, pages 126–131, 2004.

[29] Yang Xiao and Haizhon Li. Evaluation of distributed admission control for the IEEE 802.11e EDCA. *Communications Magazine, IEEE*, 42(9):S20–S24, Sept 2004.

[30] Hongqiang Zhai, Xiang Chen, and Yuguang Fang. A call admission and rate control scheme for multimedia support over IEEE 802.11 wireless LANs. In *Qshine*, pages 76–83, Jan 2004.