

# An 802.11 based MAC Protocol for Providing QoS to Real Time Applications

Mayank Mishra and Anirudha Sahoo  
Department of Computer Science and Engineering  
Indian Institute of Technology, Bombay, Powai, Mumbai, India 400076.  
email: {mayank, saho} @it.iitb.ac.in

## Abstract

IEEE 802.11 based wireless LANs (WLAN) are ubiquitous nowadays. Running real time voice and video applications over LANs is becoming common place. These applications require QoS in terms of delay, throughput etc. But 802.11 does not have inherent QoS support. Since 802.11 has a large installation base providing QoS in 802.11 WLANs is an important issue. In this paper, we propose a MAC protocol based on 802.11 which can provide QoS to real time applications. The MAC assigns different contention window to two priority classes to provide service differentiation. When collision occurs, contention window is increased in a linear fashion and the new contention windows for high and low priority traffic become non-contiguous. This unique method of contention window management provides better relative performance between the two classes. We present an analytical model to show that high priority class gets better service and report our simulation results to compare performance of our protocol with 802.11e. Further, our protocol can be implemented with very minimal change in 802.11 code.

## 1. Introduction

Real time applications involving voice or video transmissions over a network have stringent requirements in terms of delay, bandwidth and other QoS parameters. Hence, QoS should be provided by the underlying network for proper functioning of those applications. One way to achieve this is to provide QoS at the MAC layer, which makes physical bandwidth usable. Since wireless LANs (WLAN) are very common nowadays, users expect these real time applications to run over WLANs and get the required QoS. But IEEE 802.11, which is the most prevalent WLAN technology, does not have any inherent QoS support.

The IEEE 802.11 [2] MAC uses DCF (Distributed Coordination Function) for media access among the participating network nodes. But DCF alone is neither capable nor suitable for fulfilling the QoS requirements of realtime applications like voice and video. It does not provide any priority and there is no service differentiation between different flows. Generally, the proposed QoS schemes which are based on IEEE 802.11 try to improve DCF functionality. There are primarily three ways in which QoS is provided by modifying DCF based MAC:

- Prioritization among different classes of traffic: Most of the techniques use different Inter Frame Space (IFSs) or different Contention Window (CWs) or both [14, 13, 7, 10].
- Resource allocation to prioritized classes of data: This is achieved by some distributed variant of Weighted Fair Queuing (WFQ) [15, 11].
- Admission control: QoS is provided by measurement and model based admission control [16, 8, 12].

The 802.11e [3] task group was formed to come up with a priority based CSMA/CA scheme to provide differentiated services across different types of applications. The IEEE 802.11e

MAC employs a channel access function, called Hybrid Coordination Function (HCF), which includes a contention based channel access known as Enhanced Distributed Channel Access (EDCA) and a contention free channel access mechanism. EDCA has four Access Categories (ACs). Each AC obtains a differentiated channel access due to varying amount of time an AC would sense the channel to be idle and different length of the contention window size during backoff. EDCA supports eight different priorities, which are further mapped into four ACs. Access Categories are achieved by differentiating the arbitration interframe space (AIFS), the initial window size, and the maximum window size. For the AC $[i]$  ( $i = 0, \dots, 3$ ), the initial backoff window size is  $CW_{min}[i]$ , the maximum backoff window size is  $CW_{max}[i]$ , and the arbitration interframe space is AIFS $[i]$ . Each AC acts as an independent virtual MAC entity and performs the same DCF function, with a different interframe space (AIFS $[i]$ ), and a different Contention Window. Each AC has its own backoff counter (BO $[i]$ ), which is independent of others. If more than one AC finishes the backoff at the same time, the highest priority AC frame is chosen for transmission by the virtual collision handler. Other lower priority AC frames go to the next round of backoff.

In this paper we propose a MAC protocol based on 802.11 which can provide good performance for real time applications with very minimal changes to 802.11 implementation. It provides priority to real time flows by using contention window based service differentiation method. In keeping with minimal change philosophy, we do not propose to have differentiation using different IFSs. IFS based scheme provides very effective service differentiation [5]. This is because such protocols get deterministic differentiation in access times of different priority classes, whereas in CW based schemes, differentiation is probabilistic. Since 802.11e employs both IFS based and CW based priority it is quite effective in providing service differentiation. Our protocol is also carefully designed to provide very effective service differentiation among different applications. We also provide an analytical model of our protocol. Analysis of our model provides insight into the relative performance of the two priority classes. Specifically, it proves that, in saturation condition, high priority class gets more service than low priority class. We have also run simulation experiments in to compare our protocol with 802.11e.

## 2. Overview of our protocol

### 2.1. Basic Mechanism

Our protocol is designed to provide two levels of priorities. High priority can be used by real time applications like voice and video, whereas low priority would be used by regular best effort based application like email, FTP etc. Our protocol uses contention window based differentiation mechanism to provide

priorities to traffic flows. Basically, it specifies two different contention window ( $CW$ ) ranges for two priority levels. The high priority class occupies the lower half of the Contention Window range, whereas the low priority occupies the upper half.

Higher priority class chooses its backoff from the lower half of the complete Contention Window range. This allows higher priority traffic to have a smaller backoff interval than the lower priority traffic. Thus, the average delay of high priority traffic should be less than that of low priority traffic. Moreover, since the delay is low for higher priority class, it receives relatively higher throughput than the lower priority class. Thus, this MAC protocol basically provides a better quality of service to the higher priority class at the cost of service of the lower priority class.

## 2.2. Collision Handling

In IEEE 802.11 DCF MAC protocol, when a collision occurs, the contention window range is doubled. The stations involved in collision then choose the backoff value from the larger range, which lowers the probability of collision. In our scheme, however, the CW range is increased in a linear fashion. Thus, after every unsuccessful transmission attempt the current CW range is increased by  $CW_{min}$ . Further, while the CW range is increased, the individual CW ranges of priority classes become noncontiguous as shown in Figure 1. This Figure shows the effect of collision on the contention window of individual priority classes. Before collision, the contention window of high priority flow is from A to B and of low priority is from B to C. After collision, the contention window of both the priority classes become noncontiguous i.e., high priority class gets CW range from A to B and from C to D, whereas low priority class gets CW range from B to C and from D to E.

Now we provide the rationale behind two major aspects of our protocol.

### 2.2.1. Linear Increase

In IEEE 802.11 and IEEE 802.11e the increase in CW size after collision is exponential. This decreases the probability of further collision between the same stations. But in our scheme, we increase the CW size linearly. It has been reported that the probability of stations going through four or more successive collision is negligible [6]. Even the probability of having three successive collision is quite low. Moreover, the first two rounds of backoff in exponential and linear increase scheme will have the exact same contention window size. Hence, the performance difference between the two schemes may not be that significant. Linear increase in contention window size helps reducing the delay difference between packets sent from different rounds of backoff, while reducing the probability of collision in subsequent round.

### 2.2.2. Noncontiguous Contention Windows

In our protocol, contention windows of the two priority classes become noncontiguous after collision. This has the following advantages:

- Under high load condition, if a high priority and a low priority frame have collided, then they will not collide again.
- The difference between expected values of backoff for high priority and low priority remains same irrespective of the round of retransmission. This has the nice property that the relative delay performance of the two classes will still

remain the same in subsequent backoff rounds. This fact is illustrated in Figure 1 by  $exp_H$  and  $exp_L$ .

## 2.3. Management of Contention Window

As mentioned earlier, we provide service differentiation based on contention window assigned to two priority classes. Apart from this, our protocol is very similar to IEEE 802.11 DCF MAC protocol. Let  $CW_i$  denote the total contention window size in a backoff round  $i$ . When  $i = 0$ ,  $CW_i = CW_0$  is the minimum total contention window size. For this study, we have taken it as 32, which is the default for IEEE 802.11 DCF.

Since we increase total contention window linearly,  $CW_i$  is given by

$$CW_i = (i + 1) * CW_0 \quad (1)$$

Note that this  $CW_i$  is the size of the total CW range. This is divided into individual ranges of the two priority classes. Now consider the  $i^{th}$  backoff round. The total CW range is  $CW_i$  and the non-contiguous contention window for the high priority class, denoted by  $CW_i^H$  is given by

$$CW_i^H = \begin{cases} 0 & \text{to } \frac{1}{3}CW_0 - 1 \\ CW_0 & \text{to } \frac{2}{3}CW_0 - 1 \\ 2CW_0 & \text{to } \frac{5}{3}CW_0 - 1 \\ \vdots & \vdots \\ iCW_0 & \text{to } \frac{2i+1}{2}CW_0 - 1 \end{cases} \quad (2)$$

Similarly, the non-contiguous range of contention window for low priority class  $CW_i^L$  is given by

$$CW_i^L = \begin{cases} \frac{1}{3}CW_0 & \text{to } CW_0 - 1 \\ \frac{2}{3}CW_0 & \text{to } 2CW_0 - 1 \\ \frac{5}{3}CW_0 & \text{to } 3CW_0 - 1 \\ \vdots & \vdots \\ \frac{2i+1}{2}CW_0 & \text{to } (i + 1)CW_0 - 1 \end{cases} \quad (3)$$

## 2.4. Architecture

In terms of architecture a station maintains two queues, one for high priority class and the other for low priority class. So, a station has to resolve the contention between two packets belonging to the two priority classes and decide which packet should go out first. This process is termed as *virtual collision*. The winner packet would then compete with winner packets of other stations in the WLAN.

### 2.4.1. Virtual Collision Mechanism

The Virtual Collision Mechanism (VCM) resolves collision between the packets at the head of the two queues. Figure 4 explains the virtual collision process. When a frame comes to the head of the queue, its backoff value is calculated according to the non-contiguous backoff mechanism described earlier. Then the *total backoff* duration is calculated by adding *DIFS* time to the backoff value. If the two frame at the head of the two queues have the same total backoff duration, then a virtual collision would occur. In this case, it would recalculate the backoff value. But, unlike 802.11e, the contention window size does not change, while calculating the backoff value after virtual collision. The backoff value of the two packet are chosen from the non-overlapping range of the current CW. When backoff values are recalculated, choosing backoff values from the same current CW still guarantees that there will not be a virtual collision again, due to the non-overlapping nature of the CWs of high and low priority classes. Whereas in 802.11e CW size has to be doubled so that the probability of virtual collision is

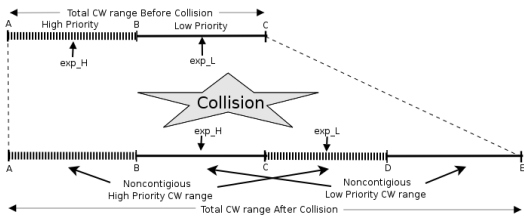


Figure 1: Effect of Collision on the Contention Window Ranges for Different Priority Classes

less in the next round, since the CWs of different Access Categories overlap. Note that virtual collision is resolved by merely recalculating the backoffs, rather than waiting for the backoff counter to count down to zero and at the end, finding out that there is a virtual collision. This saves the time which would have been wasted if the backoff countdown was to happen. If there is no virtual collision, then a winner packet goes through *backoff countdown* process. This process is very similar to DCF of 802.11. Every time, the medium is idle for a slot time, the backoff counter is decremented by one. When the medium becomes busy, the count down process is stopped. The count down starts again when the medium is sensed idle for DIFS amount of time. In Figure 4, to keep the flowchart simple, we have not shown the case where there is no need of going into backoff when the medium is free for DIFS amount of time.

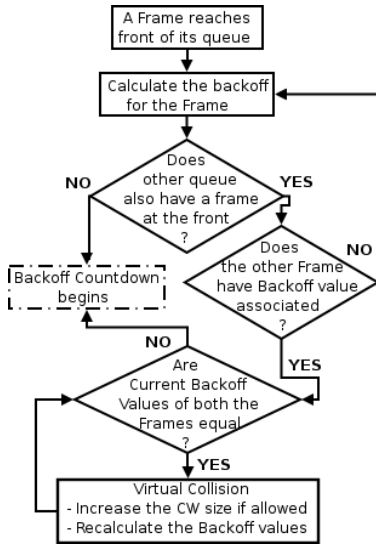


Figure 4: Flowchart Showing the Mechanism of Virtual Collision

### 2.4.2. Implementation

We introduce a thin sub-layer in the MAC layer called Virtual Collision Sub-layer (VCS) which handles virtual collisions described before. The MAC layer needs to inform VCS about events like the medium idle for DFS time, external collision, medium idle for one slot time etc. The VCS sub-layer performs backoff management and hands over frames to MAC which just sends the frame to PHY layer immediately. The current 802.11 MAC will require minimal change and the complex functional-

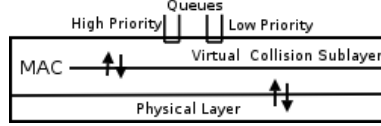


Figure 2: System Diagram of Implementation of Our Protocol

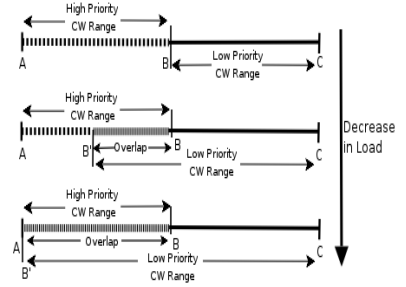


Figure 3: Variation in the size of Contention Window Range of Low priority with Change in Network load

ity will be handled by VCS sub layer. Figure 2 shows the architecture of our implementation in OPNET simulator. Only about fifty lines of code change was required to realize the protocol in OPNET.

The main functions of the MAC layer in our protocol are:

- Accept a frame from the VCS sub-layer and send it to PHY layer right away. There will not be any backoff performed by this MAC layer. Backoff management will be performed by VCS sub-layer.
- Events like medium busy, medium idle for one slot time, medium idle for DFS time, external collision are communicated to VCS sub-layer. These events are required by VCS sub-layer to do proper backoff management.

### 2.5. Overlapping Contention Window

Keeping the CWs of high and low priority classes non-overlapping leads to high delay of lower priority traffic. At light load condition, there will be very few collisions. Hence, in such condition, allowing overlap of CW of low priority with high priority improves the performance of low priority traffic without affecting the performance of high priority significantly. Since high priority flow is delay sensitive, we do not allow the same encroachment of high priority CW into low priority CW. The dynamic size of the CW of low priority class depends on the number of collisions experienced, normalized by number of transmission attempts in an observation period. The minimum and maximum size possible are  $CW/2$  and  $CW$  respectively. Let overlap factor be denoted by  $\Delta$ . This  $\Delta$  can be defined as the amount of CW which the low priority class encroaches into the CW of higher priority. Thus we have,

$$0 \leq \Delta \leq \frac{CW_0}{2} \quad (4)$$

The non-contiguous range for low priority class  $CW_i^L$  with this enhancement is given by

$$CW_i^L = \begin{cases} \frac{1}{2} \frac{CW_i}{i} - \Delta & \text{to } \frac{CW_i}{i} - 1 \\ \frac{1}{2} \frac{CW_i}{i} - \Delta & \text{to } 2 * \frac{CW_i}{i} - 1 \\ \frac{1}{2} \frac{CW_i}{i} - \Delta & \text{to } 3 * \frac{CW_i}{i} - 1 \\ \dots \\ \frac{2i-1}{2} \frac{CW_i}{i} - \Delta & \text{to } i * \frac{CW_i}{i} - 1 \end{cases} \quad (5)$$

The overlapping factor  $\Delta$  depends on the number of collisions in the network. Let  $k$  be the number of transmission attempts,  $c$  be the number of collisions in last  $k$  attempts. The overlapping factor is related inversely to the value of  $f = c/k$ . When  $f = 0$ , there is complete overlap, i.e.  $\Delta = CW_0/2$ . As  $f$  increases,  $\Delta$  decreases and finally  $\Delta$  becomes 0 when  $f \geq t$  ( $t$  is a configurable system parameter, called *collision threshold*).

### 3. Analytical Model of our Protocol

In this section, we present analytical model of our protocol. Using this model we show that higher priority flow gets better service than low priority flows. Most of the terminology used and the flow of presentation is similar to [4]. The model is Contention Window Based Differentiation Mechanism for providing QoS in Wireless LANs analyzed under saturation throughput and ideal channel conditions.

Let  $b^H(t)$  be the stochastic process representing the backoff time counter for the higher priority class in a given station and  $s^H(t)$  be the stochastic process representing the backoff stage ( $0 \dots m$ ) of the same class.

In our model, we assume, regardless of the number of re-transmissions, each packet collides with constant and independent probability  $p$ .  $p$  is the *conditional collision probability* of collision seen by a packet being transmitted. Note that this conditional probability  $p$  will be same for both the priority classes since there is no discrimination among the classes for collision once their transmission starts.

We can model the two dimensional process  $\{s^H(t), b^H(t)\}$  with the discrete-time Markov chain of higher priority depicted in Figure 5. In this Markov chain, the only non null one-step transition probabilities for high priority class are

$$\begin{aligned}
 P\{i, k|i, k+1\} &= 1 & k \in (0, W_i - 2), i \in (0, m) \\
 P\{0, k|i, 0\} &= 2(1-p)/W_0 & k \in (0, \frac{W_0}{2} - 1), i \in (0, m) \\
 P\{0, k|i, 0\} &= 0 & k \in (\frac{W_0}{2}, W_0 - 1), i \in (0, m) \\
 P\{i, k|i-1, 0\} &= 2p/W_i & k \in (j * \frac{W_i}{i+1}, (j + \frac{1}{2}) * \frac{W_i}{i+1}) \\
 & & i \in (1, m), j \in (0, i) \\
 P\{i, k|i-1, 0\} &= 0 & k \in ((j + \frac{1}{2}) * \frac{W_i}{i+1}, (j + 1) * \frac{W_i}{i+1}) \\
 & & i \in (1, m), j \in (0, i) \\
 P\{m, k|m, 0\} &= 2p/W_m & k \in ((j * \frac{W_m}{m+1}, (j + \frac{1}{2}) * \frac{W_m}{m+1}) \\
 & & j \in (0, m) \\
 P\{m, k|m, 0\} &= 0 & k \in ((j + \frac{1}{2}) * \frac{W_m}{m+1}, (j + 1) * \frac{W_m}{m+1}) \\
 & & j \in (0, m)
 \end{aligned} \tag{6}$$

The above mentioned transition probabilities represent the noncontiguous nature of the backoff mechanism of our protocol. The first transition probability in (6) accounts for the fact that at the beginning of each slot time, the backoff time is decremented by one. The second probability signifies that a new packet following a successful packet transmission starts with backoff stage 0, and thus the backoff is uniformly distributed between  $(0, \frac{W_0}{2} - 1)$ . Note that the range  $(\frac{W_0}{2}, W_0 - 1)$  is excluded from the backoff window. The third, fifth and the last probabilities in (6) captures the non-contiguity of backoff mechanism. Fourth and sixth transition probability in (6) model the system after an unsuccessful transmission. In the fourth transition probability, when an unsuccessful transmission occurs at backoff stage  $i - 1$ , the backoff goes to the next stage and the new initial backoff value is uniformly chosen in the noncontiguous range  $(j * \frac{W_i}{i+1}, (j + \frac{1}{2}) * \frac{W_i}{i+1})$  where  $j \in (0, i)$ . Finally, the sixth probability shows that once the backoff stage reaches the value  $m$ , it is not increased in subsequent packet transmissions.

Let  $b_{i,k}^H = \lim_{t \rightarrow \infty} P\{s^H(t) = i, b^H(t) = k\}$ ,  $i \in (0, m), k \in (0, W_i - 1)$  be the stationary distribution of the chain. It is easy to obtain a closed form solution for this Markov chain. First, note that:

$$\begin{aligned}
 b_{i-1,0}^H p &= b_{i,0}^H \rightarrow b_{i,0}^H &= p^i b_{0,0}^H & 0 < i < m \\
 b_{m-1,0}^H p &= (1-p)b_{i,0}^H \rightarrow b_{m,0}^H &= \frac{p^m}{1-p} b_{0,0}^H
 \end{aligned} \tag{7}$$

The two states,  $b_{i-1,0}^H$  and  $b_{m-1,0}^H$ , can have only two transitions depending on whether the station has a successful transmission or a collision. If the station succeeds at the backoff stage  $(i - 1)$ , it goes to the first stage of backoff and lands in one of the backoff slots with probability of  $\frac{(1-p)}{W_0/2}$ . On the other hand, if there was a collision, it goes to the next stage of backoff (stage  $i$ ) and lands in a backoff slot with probability  $\frac{p}{W_{i-1}/2}$ .

for each  $k \in (1, W_i - 1)$ , it is

$$b_{i,k}^H = \begin{cases} \frac{W_i - k}{W_i} \cdot 2(1-p) \sum_{j=0}^m b_{j,0}^H & \text{where, } k \in (0, \frac{W_i}{2} - 1) \\ 0 & \text{where, } k \in (\frac{W_i}{2}, W_i - 1) \end{cases} \tag{8}$$

where  $i = 0$ .

$$b_{i,k}^H = \begin{cases} \left( \frac{(j + \frac{1}{2}) \frac{W_i}{i+1} - k}{\frac{W_i}{2}} + \frac{(i-j)}{i+1} \right) p b_{i-1,0}^H & \text{where, } k \in (j * \frac{W_i}{i+1}, (j + \frac{1}{2}) * \frac{W_i}{i+1} - 1) \\ \left( \frac{i-j}{i+1} \right) p b_{i-1,0}^H & \text{where, } k \in ((j + \frac{1}{2}) * \frac{W_i}{i+1}, ((j + 1) * \frac{W_i}{i+1} - 1) \end{cases} \tag{9}$$

where  $j \in (0, i), 0 < i < m$

$$b_{i,k}^H = \begin{cases} \left( \frac{(j + \frac{1}{2}) \frac{W_i}{i+1} - k}{\frac{W_i}{2}} + \frac{(i-j)}{i+1} \right) p (b_{i-1,0}^H + b_{i,0}^H) & \text{where, } k \in (j * \frac{W_i}{i+1}, (j + \frac{1}{2}) * \frac{W_i}{i+1} - 1) \\ \left( \frac{i-j}{i+1} \right) p (b_{i-1,0}^H + b_{i,0}^H) & \text{where, } k \in ((j + \frac{1}{2}) * \frac{W_i}{i+1}, ((j + 1) * \frac{W_i}{i+1} - 1) \end{cases} \tag{10}$$

where  $j \in (0, i), i = m$ . Thus, by relations (8),(9) and (10), all the values  $b_{i,k}^H$  are expressed as functions of the value and of the conditional collision probability  $p$ .  $b_{0,0}^H$  is finally determined by imposing the normalization condition as follows.

$$1 = \sum_{i=0}^m \sum_{k=0}^{W_i-1} b_{i,k}^H \tag{11}$$

$$b_{0,0}^H = \frac{4(1-p)^2}{W_0(1+p-2p^{m+1})+2(1-p)} \tag{12}$$

We can now express the probability  $\tau^H$  that a higher priority class transmits in a randomly chosen slot time. Since a transmission occurs when the backoff time counter is equal to 0, regardless of the backoff stage, we have

$$\tau_H = \sum_{i=0}^m b_{i,0}^H = \frac{b_{0,0}^H}{1-p} = \frac{4(1-p)}{W_0(1+p-2p^{m+1})+2(1-p)} \tag{13}$$

Similarly, we have found the corresponding parameter for the low priority class. But we are unable to provide the details due to space limitation. However, a detailed derivation can be found in [9].

$$\tau_L = \frac{4(1-p)}{W_0(3-p-2p^{m+1})+2(1-p)} \tag{14}$$

It can be seen that

$$\tau_H > \tau_L \quad (\text{for } p < 1) \tag{15}$$

This means that at any given slot, the probability of transmission of a higher priority packet is more than the probability of transmission of a lower priority packet. Moreover, since the

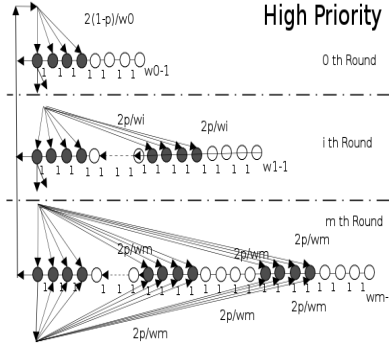


Figure 5: Markov Chain for High Priority Class

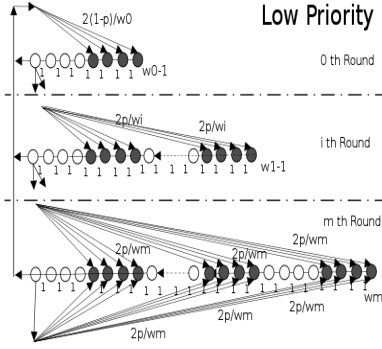


Figure 6: Markov Chain for Low Priority Class

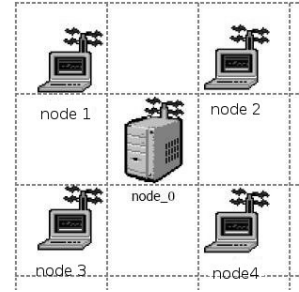


Figure 7: Simulation Topology

| Type  | AC    | IFS | $CW_{min}$ | $CW_{max}$ |
|-------|-------|-----|------------|------------|
| Voice | AC[3] | 2   | 8          | 16         |
| Video | AC[2] | 2   | 16         | 32         |
| BE    | AC[1] | 3   | 32         | 1024       |
| BG    | AC[0] | 7   | 32         | 1024       |

Table 1: 802.11e parameters

| Type  | Priority | IFS | $CW_{min}$<br>(noncontiguous) | $CW_{max}$ |
|-------|----------|-----|-------------------------------|------------|
| Voice | High     | 2   | 16                            | 1016       |
| Video | High     | 2   | 16                            | 1016       |
| other | Low      | 2   | 32                            | 1024       |

Table 2: parameters used in our protocol

| Application   | Priority | Specification      |
|---------------|----------|--------------------|
| Voice over IP | High     | 64Kbps, G.711      |
| Video Stream  | ..       | 10fr/sec, 1.38Mbps |
| FTP           | Low      | varying            |
| Telnet        | ..       | varying            |
| Database      | ..       | ..                 |
| Email, Web    | ..       | ..                 |

Table 3: Specification of Applications

probability of collision is same for both the priorities, the probability of successful transmission of the two priority classes are related as

$$\tau_H * (1 - p) > \tau_L * (1 - p) \quad (for \ p < 1) \quad (16)$$

Hence, it is obvious that the high priority class will get better service than the low priority class.

## 4. Performance Evaluation

In this Section, we present performance results of our proposed protocol in different application scenarios. We have used Opnet [1] simulation software for our experiment. Since 802.11 does not provide any priority, it is obvious that our protocol will be better than 802.11. Hence, we have compared our scheme with 802.11e only. While running our experiment for 802.11e, AC[3], AC[2] and AC[1] are used for voice, video and best effort traffic respectively, whereas AC[0] has not been used at all. As our scheme is based on 802.11, we modified the existing implementation of 802.11 in Opnet (about fifty lines of code change) to build our protocol. Performance of voice traffic is measured at low load (1.2 Mbps total offered load) scenario and at high load (3.1 Mbps total offered load) scenario. In low load scenario, only two voice flows (64 kbps each) run in the WLAN and rest is best effort flow. In high load scenario, four voice flows (64kbps each) run in the system and rest of the load is offered by best effort traffic. Performance of video traffic was only measured in one load condition (6Mbps total offered load). One video flow offered about 3Mbps load and the rest of the load is offered by best effort traffic. We could not run more than one video flow, because even with two video flows, the system ran at a very heavy load (9Mbps) resulting in heavy packet loss.

The simulation topology is shown in Figure 7. It consists of 4 wireless nodes and 1 access point. The parameters like  $CW_{min}$ ,  $CW_{max}$  and IFS are configurable in 802.11e, but for our simulation, we have taken their default values as shown in Table 1. Table 2 shows the parameter values used in our protocol. The specifications of applications and their associated priorities are given in Table 3. Different scenarios were simulated in this mode:

### 4.1. Scenario A

Following are the configuration in this scenario:

- All the four nodes have one voice application running as high priority class in high load condition, whereas in low load condition, only two nodes carry voice traffic.
- All the four nodes also run best effort traffic as low priority class in both high and low load condition.

Figure 8 and Figure 9 compare the average end-to-end delay of voice traffic in our protocol to that of 802.11e in low and high load condition. The delay of 802.11e is about 8% lower than our protocol in high load condition, but in low load condition the two have almost equal delay. This is because, at low load, there is very little collision and most of the time the medium is found free. Hence, stations seldom go into backoff. Figure 10 and Figure 11 compare throughput of voice traffic for the two protocols in low and high load conditions respectively. Since the offered load is within the capacity and the difference of average delay between the two protocol is not significant, there is no perceived difference between the throughputs. BE traffic throughput is almost the same under both the protocols both in low and high load condition (Figure 12). We have only shown the results of high load condition due to space limitation.

### 4.2. Scenario B

Following are the configurations in this scenario:

- Only one node runs video application as high priority traffic.
- All the four nodes run best effort traffic as low priority class.

Figure 13 compares the average end-to-end delay of video traffic in our protocol to that of 802.11e. Although average delay in our scheme is about 10ms more than that of 802.11e, throughput of our protocol is about 15% higher (Figure 14). Figure 15 shows the throughput of best effort traffic in this scenario. In this scenario, BE traffic throughput under our protocol is less than that in 802.11e (Figure 15). This is because, throughput of BE traffic at the node which sends both video and BE traffic is much lower than the other nodes which are carrying only BE traffic.

We also ran experiment with overlapping CW, as discussed in Section 2.5. Figure 16 compares the performance of Best Effort traffic with and without overlapping CW, when collision threshold  $t$  is 0.32. There is about 5% increase in Best Effort traffic throughput. Allowing overlap CWs increases delay of voice traffic slightly (not shown due to space limitation).

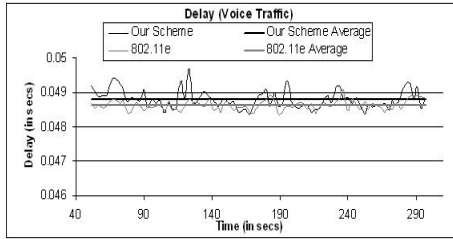


Figure 8: Average Delay of Voice (low load)

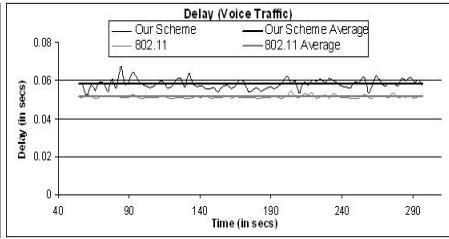


Figure 9: Average Delay of Voice (high load)

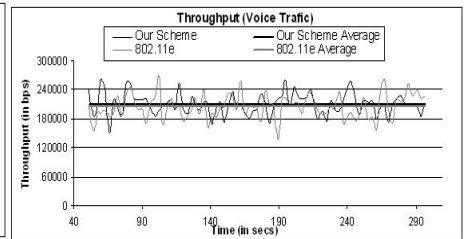


Figure 10: Average Throughput of Voice (low load)

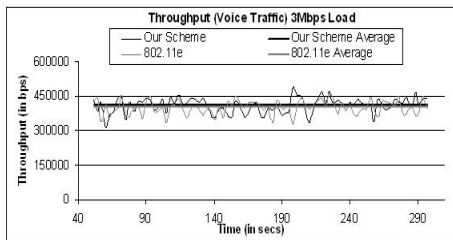


Figure 11: Average Throughput of Voice (high load)

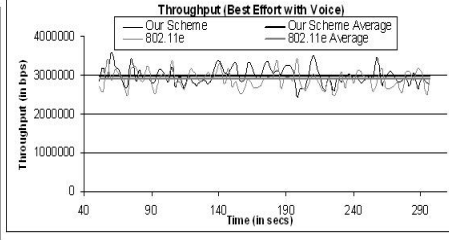


Figure 12: Throughput of BE in Scenario A (high load)

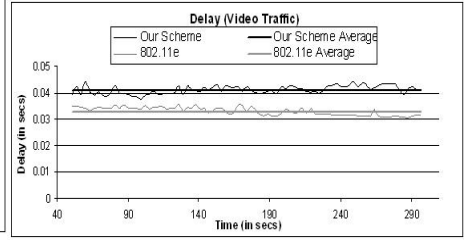


Figure 13: Average Delay of Video Traffic

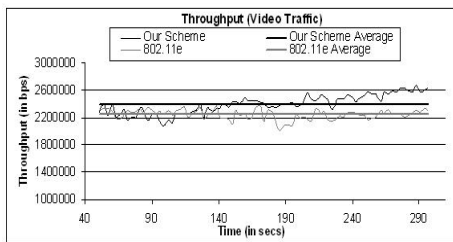


Figure 14: Average Throughput of Video Traffic

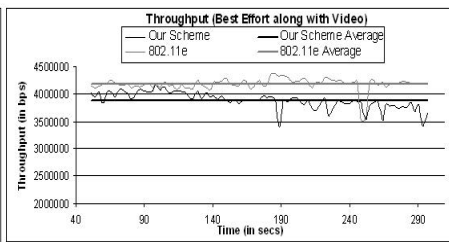


Figure 15: Throughput of BE in Scenario B

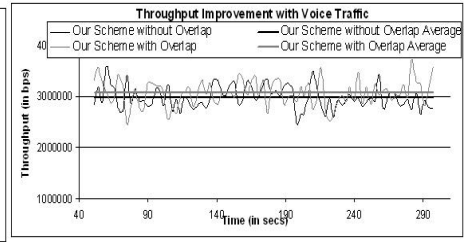


Figure 16: Throughput of Best Effort with Voice Traffic with Overlapping CW

## 5. Conclusion

We have presented an 802.11 based MAC protocol that provides QoS to real time traffic like voice and video applications. Our MAC protocol provides priority to real time traffic by using contention window based service differentiation method. We presented an analytical model to show that high priority traffic gets more service than low priority traffic in saturation condition. Our simulation experiments show that performance of our protocol is as effective as 802.11e for voice and video traffic. In summary, delay performance of high priority traffic in our protocol is quite close to that of 802.11e, throughput of high priority traffic is almost same under the two protocols. We have implemented our protocol with very minimal change to the 802.11 code in OPNET simulator. Hence, implementing our protocol in real 802.11 firmware should be easy.

## References

- [1] Opnet network simulator. <http://www.opnet.com>.
- [2] Wireless lan medium access control(mac) and physical layer (phy) specifications. IEEE 802.11 Standard, 1997.
- [3] Wireless lan medium access control(mac) and physical layer (phy) specifications. IEEE 802.11e Standard, 2005.
- [4] G. Bianchi. Performance analysis of the ieee 802.11 distributed coordination function. *IEEE Journal on Selected Areas in Communications*, 18(18), March 2000.
- [5] G. Bianchi, I. Tinnirello, and L. Scalia. Understanding 802.11 contention-based prioritization mechanisms and their coexistence with legacy 802.11 stations. *IEEE Network*, 19(4):28–34, Aug 2005.
- [6] M. Bottigliengo, C. Casetti, C. F. Chiasserini, and M. Meo. Short-term fairness for tcp flows in 802.11b wlan. *IEEE Infocom*, 2004.
- [7] J. Deng and R. Chang. A priority scheme for IEEE 802.11 DCF access method. pages 96–102. *IEICE Transactions in Communications*, 1999.
- [8] S. Garg and M. Kappes. Call admission control for 802.11. In *Proc of. GLOBECOM 2003.*, 2003.
- [9] M. Mishra and A. Sahoo. A contention window differentiation mechanism for providing qos to high priority data in 802.11. <http://www.it.iitb.ac.in/research/techreport/reports/19.pdf>, Tech. Report, KReSIT, IIT Bombay, 2006.
- [10] A. Nafaa, A. Ksentini, and A. Mehaoua. Scw: Sliding contention window for efficient service differentiation in 802.11 networks. In *Proc of. IEEE WCNC 2005, The IEEE Wireless Communication and Networking Conference.*, 2005.
- [11] W. Pattara-atokom, P. Krishnamurthy, and S. Banerjee. Distributed mechanisms for quality of service in wireless lans. pages 26–34. *IEEE Wireless Communications*, June, 2003.
- [12] D. Pong and T. Moores. Admission control for voip traffic in ieee 802.11 contention access mechanism. pages 174–178. In *Proc of. GLOBECOM 2003.*, 2003.
- [13] L. Romdhani, Q. Ni, and T. Turlletti. Adaptive EDCF: Enhanced Service Differentiation for IEEE 802.11 Wireless Ad Hoc Networks. 2004.
- [14] L. Romdhani, Q. Ni, and T. Turlletti. A survey of qos enhancements for ieee 802.11 wireless lan. 2004.
- [15] N. Vaidya, P. Bahl, and S. Gupta. Distributed fair scheduling in a wireless lan. Boston, USA, August 2000. Annual International Conference on Mobile Computing and Networking (Mobicom 2000).
- [16] Y. Xiao and H. Li. Evaluation of distributed admission control for the 802.11e edca. *IEEE Radio Communications*, pages S20–S24, Sept. 2004.