A Call Admission and Rate Control Scheme for Multimedia Support over IEEE 802.11 Wireless LANs *

Hongqiang Zhai, Xiang Chen, and Yuguang Fang
Department of Electrical & Computer Engineering
University of Florida, Gainesville, Florida 32611-6130
E-mail: zhai@ecel.ufl.edu, xchen@ecel.ufl.edu, and fang@ece.ufl.edu

Abstract

In this paper, we proposed a novel call admission and rate control (CARC) scheme. Unlike previous research works that are focused on providing service differentiation in the contention-based 802.11 DCF, we aim to support stringent QoS requirements of real-time and streaming traffic. The key idea of this scheme is to regulate the arriving traffic of the WLAN such that the network can work at an optimal point. We first show that the channel busyness ratio is a good indicator of the network status in the sense that it is easy to obtain and can accurately and timely represent channel utilization. Then we propose two algorithms that function upon the use of the channel busyness ratio. The call admission control algorithm is used to regulate the admission of real-time or streaming traffic and the rate control algorithm to control the transmission rate of best effort traffic. A comprehensive simulation study in ns-2 has verified the performance of our proposed CARC scheme, showing that the original 802.11 DCF protocol can statically support strict QoS requirements, such as those required by voice over IP or streaming video, and at the same time, achieve a high channel utilization.

1. INTRODUCTION

In recent years, the IEEE 802.11 wireless LAN [6] has been increasingly employed to access the Internet because of its simple deployment and low cost. According to the IEEE 802.11 standard, the medium access control (MAC) mechanism contains two access methods, i.e., Distributed Coordination Function (DCF) and Point Coordination Function (PCF), with the former being specified as the fundamental access method. Despite its popular use, currently only best effort traffic is supported in DCF.

* This work was supported in part by the U.S. Office of Naval Research under Young Investigator Award N000140210464 and under grant N000140210554.

Quality of service (QoS) provisioning for multimedia services including voice, video, and data is crucial for the IEEE 802.11 wireless LAN to continue to thrive and evolve as a viable wireless access to the Internet. Although there are several schemes (such as [2,12]) which use PCF mode to support QoS for real-time traffic, we do not further discuss along this line because PCF is an optional access method ([6]) which is only usable on infrastructure network configurations and not supported in most current wireless cards. In addition, it may result in poor performance as shown in [11, 20]. In this paper, we thus focus on the 802.11 DCF mode. However, guaranteeing QoS for real-time traffic in the 802.11 DCF mode is not an easy task given that it is in nature contention-based and distributed, and hence render effective and efficient control very difficult.

In the face of these challenges, considerable research ([1, 15, 17–19]) has been conducted to enhance the IEEE 802.11 WLAN to support service differentiation or prioritized service [3]. Ada and Castelluccia [1] proposed to scale the contention window, use different inter frame spacing or maximum frame length for services of different priority. As a matter of fact, similar ideas have recently been adopted in the enhanced DCF (EDCF) defined in the IEEE 802.11e draft ([7]). In [18], two mechanisms, i.e., virtual MAC and virtual source, were proposed to enable each node to provide differentiated services for voice, video, and data. Similarly, by splitting the transmission period into a real-time one and a non-real-time one, real-time traffic is supported with QoS guarantee [15]. However, the DCF mode was dramatically changed. The Blackbust in [17] provided high priority for real-time traffic. Unfortunately, it imposes special requirements on high priority traffic and is not fully compatible with the existing 802.11 standard. In summary, if the semantics of the 802.11 DCF is maintained, only differentiated service, rather than stringent QoS assurance, is supported .

In our previous work [21], We have shown through both theoretical and simulation studies that the IEEE 802.11 DCF protocol could satisfy the QoS requirements of the real-time and streaming traffic while achieving the maximal channel utilization when it is working at the optimal point corresponding to a certain amount of arriving traffic. If the arriving traffic is heavier than this threshold, the WLAN enters into saturation, resulting in significant increase in delay and decrease in throughput; on the other hand, if the arriving traffic is less than this threshold, channel capacity is wasted. In reality, however, to tune the network that operates on the basis of channel contention to work at this point requires an effective and efficient control algorithm to regulate the input traffic [14]. Therefore, we are motivated to design a call admission and rate control scheme (CARC) (Section 3). Specifically, call admission control (CAC) is used for real-time or streaming traffic, and rate control (RC) for best effort data traffic.

Essentially, the CARC scheme has the following distinguishing features:

- It utilizes an new measure of network status, the channel busyness ratio to exercise traffic regulation, which is easy to obtain and can accurately and timely represent the network utilization as shown in Section 2.
- The call admission control scheme is able to provide statistical QoS guarantees for real-time and streaming traffic.
- The rate control scheme allows best effort traffic to utilize all the residual channel capacity left by the real-time and streaming traffic while not violating their QoS metrics, thereby enabling the network to approach the maximal theoretical channel utilization.
- Since each node keeps track of the channel busyness ratio locally to conduct control, this scheme is distributed and suits well with the DCF mode.

We have implemented the CARC scheme in *ns-2* [13], and conducted a comprehensive simulation study to evaluate its performance. As shown in Section 4, CARC is able to support real-time services, such as voice and video, with QoS guarantees, and achieve high throughput by allowing best effort traffic to make full use of the residual channel capacity. This confirms that the 802.11 WLAN can not only support differentiated service, but also support strict QoS.

Finally, Section 5 concludes this paper.

2. CHANNEL BUSYNESS RATIO

In this section, we give several definitions and elaborate on why and how the channel busyness ratio can be used to represent the network status.

Channel busyness ratio R_b is the ratio of the time that the channel is determined busy to the total time. Both successful transmissions and collisions contribute to R_b. We denote R_s as *channel utilization* which is the ratio of successful transmission periods to the total time. R_s counts every period T_{suc} with a successful transmission, and T_{suc} includes the transmission time for *rts*, *cts*, *data* and *ack* and all necessary interframe spacings, i.e., *sifs* and *difs*.

Bit rate for DATA packets	2 Mbps
Bit rate for RTS/CTS/ACK	1 Mbps
PLCP Data rate	1 Mbps
Backoff Slot Time	20 μs
SIFS	10 μs
DIFS	50 μs
Phy header	192 bits
MAC header	224 bits
DATA packet	8000 bits + Phy header + MAC header
RTS	160 bits + Phy header
CTS, ACK	112 bits + Phy header

Table 1. IEEE 802.11 System Parameters

It is easy to conduct the measurement of R_b since the IEEE 802.11 is a CSMA-based MAC protocol, working on the physical and virtual carrier sensing mechanisms. The channel is determined to be busy when the measuring node is sending, receiving, or its network allocation vector (NAV) [6] indicates the channel is busy, and to be idle otherwise.

R_b can be used to represent the network status. We present some ns-2 simulation results in Fig. 1, which shows the performance of throughput, delay, and delay variation as a function of the channel busyness ratio. The IEEE 802.11 system parameters are summarized in Table 1. Every node initiates an identical UDP/CBR traffic flow to a randomly selected neighbor. The queue length at each node is 100 packets. In Fig. 1, channel busyness ratio increases with the sending rate of flows.

It can be seen that there is a turning point in all the curves, where the channel busyness ratio is about 0.95. Before that point, as the input traffic increases, the throughput keeps increasing linearly with R_b, the delay (including queueing delay, backoff time and transmission time) and delay variation does not change much and is small enough to support the real-time traffic [8, 9]. After that point, the throughput drops quickly and the delay and delay variation increase dramatically. Clearly, this turning point is the optimal operating point that we should tune the network to work around, where the throughput is maximized and the delay and delay variation are small. Also we notice that there is almost no collisions and $R_b \approx R_s$ before the turning point. Therefore, the network status is known by keeping track of the channel busyness ratio.

Further, if we denote by B_U the channel utilization corresponding to the optimal point, we can estimate the available normalized throughput by $s_a = (B_U - R_b) \times \overline{data}/T_{suc}$

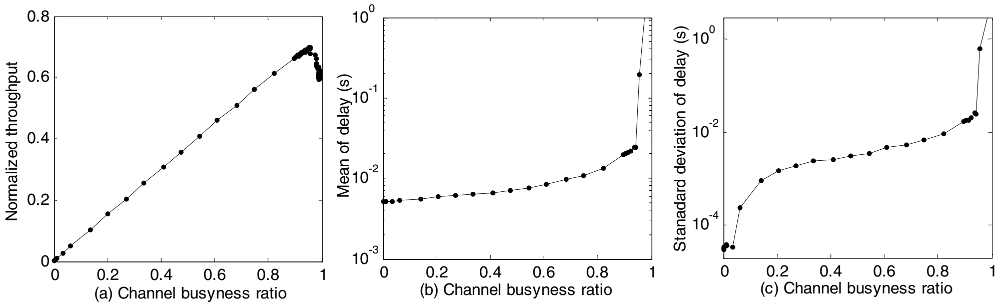


Figure 1. Simulation results when number of nodes equals 50 and RTS/CTS mechanism is used

before the network achieves the maximal throughput. As shown in [21], B_U is almost the same for different number of active nodes and packet size, and $B_U \approx 0.90$ (without RTS/CTS) or $B_U \approx 0.95$ (with RTS/CTS).

3. CARC: CALL ADMISSION AND RATE CONTROL

As revealed in previous sections, keeping the channel busyness ratio close to a certain threshold is essential to maximizing network throughput and providing QoS. To accomplish this goal, it is crucial to regulate total input traffic through call admission control (CAC) over real-time traffic and rate control (RC) over best effort traffic, given that the 802.11 DCF protocol is designed to provide best effort services and does not differentiate any types of services.

We thus propose an call admission and rate control (CARC) scheme, which consists of two mechanisms: CAC and RC. In what follows, the design rationale is discussed first, followed by detailed descriptions of the CAC and RC algorithm in order.

3.1. Design Rationale

In the context of the WLAN where each node only has a partial view of the network, the design of CARC is much more complicated than it appears, especially due to the following difficulties.

The first problem is that multiple new real-time flows may be simultaneously admitted by individual nodes if not coordinated so that the wireless LAN may be overloaded and QoS will be violated, henceforth referred to as *over-admission*.

Another more severe issue is that it is very hard for each individual node to accurately estimate the total traffic rate of the currently admitted real-time flows based on the measured channel busyness ratio, since the latter also includes the contribution from best effort traffic. Without an accurate estimate, the rate of best effort traffic cannot be effectively controlled. This in turn may completely cause the CAC algorithm to reject any real-time traffic if the channel busyness ratio is boosted to a high level by heavy best effort traffic.

Therefore, to achieve its goal, the CARC scheme must properly address these problems. To completely avoid the over-admission problem, we opt for a coordinator-aided CAC scheme. In other words, all admission decisions are made by a coordinating node, which can record the current number of admitted real-time flows and their occupied channel bandwidth in the network. Clearly, in this way no over-admission will occur. It is important to note that a coordinator is available whether the wireless LAN is working in the infrastructure mode or in the ad hoc mode. If the network is working in the infrastructure mode, the access point is the coordinator. Otherwise, a mobile node can be elected to act as the coordinator in the network using one of many algorithms in the literature ([5] [16]). Further discussions on the election algorithm is beyond the scope of this paper.

Since the 802.11 DCF is not prioritized, our CAC algorithm guarantees a uniform QoS level in terms of delay, delay variation, and packet loss rate for all real-time traffic. Note that two criteria are applied to CAC. The first criterion is that CAC admits a new real-time flow only if the requested resource is available. Here we need to set an upper bound, denoted by B_M, for bandwidth reservation for real-time traffic [4]. We set B_M to 80% (it could be changed depending on traffic composition) of the maximum channel utilization, denoted by B_U, of the WLAN for two reasons. It first ensures that the best effort traffic is operational all the time, since the best effort traffic is at least entitled to 20% of the channel throughput. In addition, the 20% of the channel throughput for best effort traffic can be used to accommodate sizable fluctuations caused by VBR real-time traffic. The second criterion is that the QoS provided for the currently existing real-time flows is not affected. This can be guaranteed as long as the first criterion is in place to make sure the collision probability is kept around a small value as shown earlier.

For best effort traffic, the rate control (RC) scheme must ensure two things. First, best effort traffic should not affect the QoS level of the admitted real-time traffic. Second, best effort traffic should have access to the residual bandwidth left by real-time traffic in order to efficiently utilize the channel. Clearly, both demand an accurate estimate of the instantaneous rate of ongoing real-time traffic. If the network is working in the infrastructure mode, this is achievable. In this case, since all communications must go through the access point, it can monitor the traffic in both directions, i.e., the upstream flows that are from mobile nodes to the access point, and the downstream flows that are from the access point to mobile nodes. On the other hand, if the network is working in the ad hoc mode, accurate rate control becomes much more difficult. In this case, since all mobile nodes can directly communicate with each other, no node has perfect knowledge of the instantaneous traffic rate of the real-time traffic, as the access point does. At the same time, no single node can accurately monitor all the traffic in the air and control the traffic rate of every other node. Therefore, an effective distributed rate control scheme is needed for the ad hoc mode.

3.2. Call Admission Control

In the CAC scheme, three parameters, (TR, TR_{peak}, len), are used to characterize the bandwidth requirement of a real-time flow, where TR is the average rate and TR_{peak} the peak rate, both in (bit/s), and len is the average packet length in bits. For CBR traffic, $TR = TR_{peak}$. For VBR traffic, $TR < TR_{peak}$. We use the channel utilization cu that a flow will occupy to describe the bandwidth requirement, and

$$cu = \mathcal{U}(TR) = \frac{TR}{len} \times T_{suc}, \qquad (1)$$

where $\mathcal{U}$ is the mapping function from traffic rate to channel utilization, and T_{suc} is defined in Section 2. Thus (cu, cu_{peak}) specify a flow's bandwidth requirement, where $cu = \mathcal{U}(TR)$ and $cu_{peak} = \mathcal{U}(TR_{peak})$.

On the side of the coordinator, the total bandwidth occupied by all admitted real-time flows is recorded in two parameters, i.e., the aggregate (cu, cu_{peak}), denoted by (cu_A, cu_{peak_A}), which are updated when a real-time flow joins or leaves through the following admission procedure.

When receiving a real-time connection request from its application layer, a node must send a request with specified (cu, cu_{peak}) to the coordinator, noting that it wants to establish a real-time flow. Only after the request is admitted, the node starts to establish the flow with the intended destination. Otherwise, the node rejects the request and informs the corresponding application.

Upon receiving a QoS request with parameters (cu, cu_{peak}), the coordinator checks whether the remainder of the quota B_M can accommodate the new real-time flow. Specifically, it carries out the following:

- If $cu_A + cu < B_M$ and $cu_{peak_A} + cu_{peak} < B_U$ [1], the coordinator issues the "connection admitted" message, and updates (cu_A, cu_{peak_A}) accordingly;
- Otherwise, the coordinator issues the "connection rejected" message.

Finally, when a real-time flow ends, the source node of the flow should send a "connection terminated" message to the coordinator, and the latter responds with a "termination confirmed" message and updates (cu_A, cu_{peak_A}) accordingly.

Note that real-time packets have highest priority in the outgoing queue, which means they will always be put on the top of the queue. Meanwhile, all the control messages related to connection admission and termination are transmitted as best effort traffic; however, they have higher priority than other ordinary best effort packets, which have the lowest priority. By doing so, we make sure that these messages do not affect the real-time traffic while being transmitted promptly.

3.3. Rate Control

3.3.1. Rate control in infrastructure mode We adopt a sliding window smoothing algorithm to estimate the aggregate instantaneous bandwidth requirement of the real-time traffic cu_{Ar}. Let us denote by ti_{int} the period between the $(i-1)$-th and i-th successful packet transmission or reception at the access point, and denote by ti_{real} the time consumed by real-time traffic in this period. Apparently, if the i-th packet is a TCP packet, $ti_{real} = 0$. Thus we have

$$cu_{Ar_i} = \sum_{j=i+1-k}^{i} ti_{real} \Big/ \sum_{j=i+1-k}^{i} ti_{\text{int}}, \qquad (2)$$

where k is the sliding window size. Thus the instantaneous available bandwidth for best effort traffic, denoted by cu_{bi}, is

$$cu_{bi} = B_U - cu_{Ar_i} \qquad (3)$$

If the recent k packets are all TCP packets, then $cu_{Ar_i} = 0$ and all the bandwidth will be allocated to TCP flows. Once a real-time packet which has higher priority in the outgoing queue is transmitted or received, the rate of TCP flows will be decreased. This algorithm thus effectively adapts TCP

1 Note that this criterion can provide QoS guarantees for VBR real-time traffic, although it is conservative if cu_{peak_A}/cu_A is much larger than B_U/B_M. This problem could be alleviated if we use measured values of cu_A or cu_{peak_A}; however, it is well known that when the number of present real-time flows is small, the CAC must also be conservative in order not to cause serious QoS degradation [10]. We will further investigate this issue in our future work.

rate to the change of VBR traffic rate. Clearly, if k is small, the estimation is aggressive in increasing TCP rate; if k is large, the estimation is conservative [10]. We set k to 10 in our simulation as a tradeoff.

Given cu_b, the task is to fairly allocate the bandwidth to all the nodes that have the best effort traffic to transmit. We assume the number of nodes that are the sources of downstream flows is n_d, and the number of nodes that are the sources of upstream flows is n_u. Obviously, the access point knows both n_d and n_u. Thus the traffic rate for the best effort traffic allocated to the access point TR_{ba} and that allocated to each mobile node TR_{bm} are as follows.

$$\begin{array}{l} TR_{ba} = \mathcal{U}^{-1}(cu_b \times n_d/(n_u + n_d)) \\ TR_{bm} = \mathcal{U}^{-1}(cu_b/(n_u + n_d)) \end{array}, \tag{4}$$

where $\mathcal{U}^{-1}$ is the inverse function of $\mathcal{U}$ defined in Equation (1).

This rate allocation TR_{ba} immediately takes effect at the access point. And the rate allocation TR_{bm} is piggybacked to each mobile node by using the MAC layer ACK frame for each best effort packet from the node. In this way, the mobile node can immediately adjust the transmission rate of its own best effort traffic. Two bytes needs to be added in the ACK frame to indicate TR_{bm} with a unit of $R_D \times 2^{-16}$, where R_D is the bit rate for the MAC layer DATA packets.

Note that the above fair allocation algorithm is only one choice for rate control. Depending on traffic patterns, other allocation algorithms can also be used, since the access point can monitor the instantaneous rate of each best effort flows from/to each mobile node. For instance, it is easy to design an algorithm that allocates different rate to different flows by modifying Equation (4).

3.3.2. Rate control in ad hoc mode We propose a novel, simple and effective rate control scheme for the best effort traffic at each node. In this scheme, each node needs to monitor the channel busyness ratio R_b during a period of T_{rb}. Let us denote by R_{br} the contribution from real-time traffic to R_b, and denote by TR_b the traffic rate of best effort traffic at the node under consideration, with the initial value of TR_b being conservatively set, say one packet per second. The node thus adjusts TR_b after each T_{rb} according to the following:

$$TR_{b_{new}} = TR_{b_{old}} \times \frac{R_{bt} - R_{br}}{R_b - R_{br}}, \tag{5}$$

where $TR_{b_{new}}$ and $TR_{b_{old}}$ are the value of TR_b after and before the adjustment, and R_{bt} is a threshold of channel busyness ratio and is set to $95\% \times B_U$. Two points are noted on Equation (5). First, we see that the node increases the rate of best effort traffic if $R_b < R_{bt}$ and decreases the rate otherwise. Second, if all the nodes adjust the rate of its own best effort traffic according to Equation (5), the total best effort traffic rate will be

$$\sum TR_{b_{new}} = \sum TR_{b_{old}} \times \frac{R_{bt} - R_{br}}{R_b - R_{br}} \approx \mathcal{U}^{-1}(R_{bt} - R_{br}), \tag{6}$$

where $\sum TR_{b_{old}} \approx \mathcal{U}^{-1}(R_b - R_{br})$ is due to the fact that $R_s \approx R_b$ as shown in Section 2 and $R_b - R_{br}$ is the contribution from the total best effort traffic to R_b. Thus after one control interval T_{rb}, the channel utilization will approximately amount to R_{bt}.

Apparently this scheme depends on the estimation of R_{br}. Larger estimate of R_{br} results in larger increase in traffic rate when $R_{bt} > R_b$ and larger decrease in traffic rate when $R_{bt} < R_b$. On the contrary, smaller estimate of R_{br} results in smaller increase in traffic rate when $R_{bt} > R_b$ and smaller decrease in traffic rate when $R_{bt} < R_b$. To avoid overloading the wireless LAN and protect the QoS level of admitted real-time traffic, a conservatively increasing and aggressively decreasing law is desired for controlling the best effort traffic rate. This is especially preferred given the fact that an accurate estimate of R_{br} is not available. These considerations have led us to the following scheme to estimate R_{br}.

Each mobile node needs to monitor all the traffic in the air. Note that, to be consistent with the original 802.11 protocol, our scheme only requires mobile nodes to decode the MAC header part, as the original 802.11 does in the NAV procedure, instead of the whole packet it hears. For the purpose of differentiating real-time packets from best effort packets, one reserved bit in the subtype field of the MAC header is used. Therefore, the observed channel busyness ratio comprises three pieces of contribution: the contribution from best effort traffic with a decodable MAC header R_{b1}, that from real-time traffic with a decodable MAC header R_{b2}, and that of all the traffic with an undecodable MAC header R_{b3} due to collision. So we give an upper bound, a lower bound, and an approximation for R_{br} as follows:

$$\begin{array}{l} R_{b2} \leqslant R_{br} \leqslant R_{b2} + R_{b3} \\ R_{br} \approx R_{b2} \times (1 + \frac{R_{b3}}{R_{b1}+R_{b2}}) = \frac{R_{b2} \times R_b}{R_{b1}+R_{b2}} \equiv \hat{R}_{br} \end{array}, \tag{7}$$

where we assume R_{b3} is composed of real-time traffic and best effort traffic according to the ratio of R_{br}/R_b.

To enforce a conservatively increasing and aggressively decreasing law, we thus set R_{br} as follows:

$$R_{br} = \begin{cases} R_{b2}, & if\ R_b < R_{bt}; \\ R_{b2} + R_{b3}, & if\ R_b > R_{bt}. \end{cases} \tag{8}$$

We also need to determine the control interval T_{rb} distributedly. To be responsive to the change of the channel busyness ratio observed in the air, the rate is adjusted at each time instant when a node successfully transmits a best effort packet. Thus T_{rb} is set to the interval between two successive best effort packets that are successfully transmitted. Note that even when such an interval is short and thus no real-time traffic is observed in it, i.e., $R_{br} = 0$, the rate of best effort traffic can at most be increased to $\mathcal{U}^{-1}(R_{bt})$. At that time, the collision probability is still very small according to previous analysis, so the real-time packets later on can be quickly transmitted, which will in turn lower the best effort traffic rate.

4. PERFORMANCE EVALUATION OF CARC

We have implemented the CARC scheme in *ns-2* simulator [13]. In this section, we evaluate its effectiveness in an 802.11 wireless LAN.

4.1. Simulation Configuration

An 802.11 based wireless LAN with 100 mobile nodes is simulated. In all simulations, channel rate is 2 Mb/s and simulation time is 120 seconds. The queue length at each node is 100 packets. The IEEE 802.11 system parameters are summarized in Table 1

To model multimedia traffic, three different classes of traffic are considered:

Voice Traffic (VBR): The voice traffic is modeled as VBR using an *on/off* source with exponentially distributed *on* and *off* periods of 300 ms average each. Traffic is generated during the *on* periods at a rate of 32 kb/s with a packet size of 160 bytes, thus the inter-packet time is 40 ms.

Video Traffic (CBR): The video traffic is modeled as CBR traffic with a rate of 64 kb/s with a packet size of 1000 bytes, thus the inter-packet time is 125 ms.

Data Traffic Model (UBR): We use the greedy best-effort TCP traffic as the background data traffic with a packet size of 1000 bytes.

During simulation, the RTS/CTS mechanism is used for video and TCP packets, but not used for voice packets because of its relatively large overhead. The traffic load is gradually increased, i.e., a new voice, video or TCP flow is periodically added in an interleaved way, to observe how CARC works and the effect of a newly admitted flow on the performance of previously admitted flows. Specifically, until 95 second, a new voice flow is added at the time instant of $6 \times i$ second $(0 \leqslant i \leqslant 15)$. Likewise, a video flow is added two seconds later and a TCP flow is added 4 seconds later. Furthermore, to simulate the real scenario where the start of real-time flows are randomly spread over time, the start of a voice flow is delayed a random period uniformly distributed in [0ms, 40ms], and that of a video flow delayed a random period uniformly distributed in [0ms, 125ms]. Note that in the simulation period between [95s, 120s], we purposely stop injecting more flows into the network in order to observe how well CARC performs in a steady state.

Two scenarios shown below are investigated.

Infrastructure Mode: In this case, all flows pass through the access point, whereby half number of flows are downstream, and another half are upstream. The source or the destination of these flows which are not the access point is randomly chosen from all the mobile nodes other than the access point.

Ad Hoc Mode: In this case, there is no fixed access point. Therefore, the source and the destination of all flows are randomly chosen from all the mobile nodes. All the other parameters are the same as those in the infrastructure mode.

4.2. Simulation Results

From the simulation results, we find there are a total of 12 voice flows and 11 video flows admitted at 66 second; and no more voice or video flows are admitted thereafter. The number of TCP flows increases by one every 6 s until 95 second. After 95 s, as expected, there is no change in the number of flows.

As can be calculated using Equation (1), each voice flow contributes 0.0347 to the channel busyness ratio R_b, and each video flow 0.04339 by noticing that each packet is added a 20 bytes IP header in *ns-2*. Thus after 12 voice and 11 video flows are admitted, the portion of R_b that accounts for the voice flows is $0 \sim 0.38$, with a mean of 0.19, and the portion that accounts for the video connections is 0.52. Thus $\mathcal{U}(TR_A) = 0.71$, and $\mathcal{U}(TR_{Apeak}) = 0.90$. Thereafter, the admission control mechanism starts to reject the following real-time flows.

4.2.1. Infrastructure mode Fig. 2(a) shows the throughput for the three traffic types throughout the simulation. At the beginning, the TCP traffic has high throughput; then as more real-time flows are admitted, it gradually drops as a result of rate control. Because we set an upper bound B_M for real-time traffic, it can be observed that when the traffic load becomes heavy, TCP traffic, as desired, is not completely starved. Because TCP traffic is allowed to use any available channel capacity left by real-time traffic, the total channel throughput, namely the sum of the throughput due to different types of traffic, always remains steadily high. Note that the throughput for the TCP traffic does not include the contribution from TCP ACK packets, even though they also consume channel bandwidth to get through. Thus, the total channel throughput should be somewhat higher than the total throughput as shown in Fig. 2(a).

The end-to-end delay is illustrated in Fig. 2(b), in which every point is averaged over 2 seconds. It can be observed

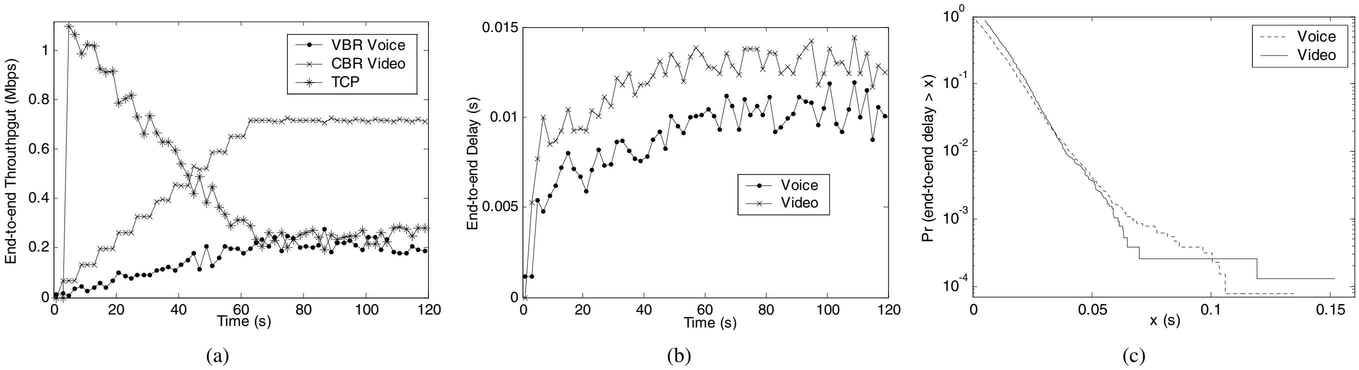


Figure 2. Infrastructure mode: the number of real-time and TCP flows increases over time.

	mean	SD^1	$97\%ile^2$	99 %ile	99.9 %ile
Voice	0.0097	0.0089	0.0306	0.0412	0.0670
Video	0.0127	0.0081	0.0314	0.0392	0.0609
1: standard deviation; 2: percentile.					

Table 2. Statistics in the Infrastructure Mode.

that the delay for real-time traffic is always kept below 20 ms. Initially, as the number of admitted real-time flows increases, the delay increases. Note that the increase of delay is not due to TCP traffic, but due to the increasing number of competing real-time flows. Then, the delay oscillates around a stable value. Fig. 2(c) presents the delay distribution for voice and video traffic. More detailed statistics of delay and delay variation is given in Table 2. As shown in Table 2, the 97 percentile delay value for voice and video is 35.5 ms and 32.2 ms respectively, and the 99 percentile delay value for voice and video is 55.4 ms and 45.2 ms respectively. It is known that for real-time traffic, packets that fail to arrive in time is simply discarded. Given the allowable $1\% \sim 3\%$ packet loss rate, these delays are well within the bounds given in the standards [8,9]. The good delay performance indicates that the CARC scheme can effectively guarantee the delay and delay jitter requirements of real-time traffic, even in the presence of highly dynamic TCP traffic.

Finally, we note that in simulation, no lost real-time packet is observed. This should be accredited to the fact that our CARC scheme successfully maintains a very low collision probability, thereby avoiding packet losses due to collisions. Also, since the network is tuned to work in the optimal point, no real-time packet is lost due to buffer overflow.

4.2.2. Ad hoc mode Fig. 3 illustrates the performance of the CARC scheme when it works in the ad hoc mode. Again, the performance is very good. The CARC scheme

	mean	SD	97 %ile	99 %ile	99.9 %ile
Voice	0.0101	0.0104	0.0350	0.0500	0.0876
Video	0.0133	0.0092	0.0337	0.0477	0.0903

Table 3. Statistics in the Ad Hoc Mode.

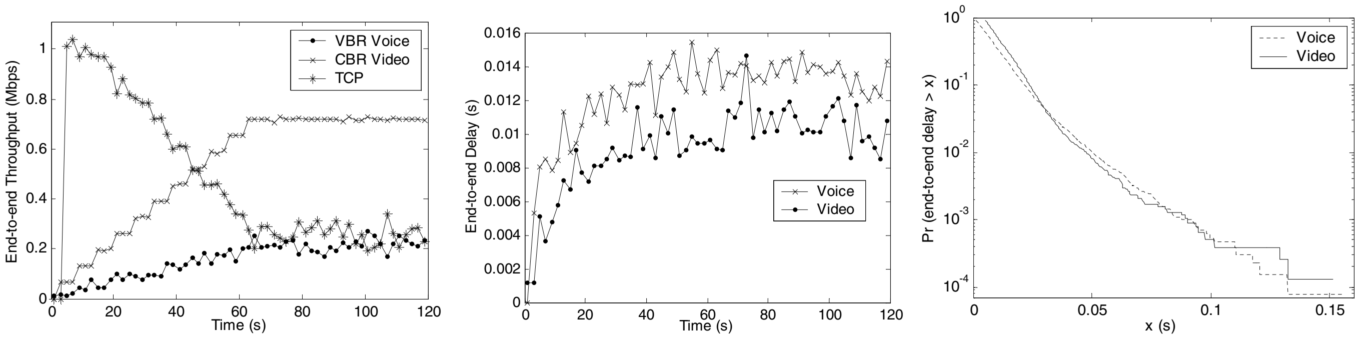
delivers almost the same throughput and average end-to-end delay, and also no lost real-time packet is observed. However, as seen from Fig. 3(c), the delay variation is slightly larger, which is also confirmed in Table 3. This is due to the imperfect estimation of the rate of real-time traffic in the ad hoc mode, as each node locally estimates the rate.

In conclusion, the simulation results demonstrate our CARC scheme performs well when the network operates either in the infrastructure mode or in the ad hoc mode. Consequently, the strict QoS of real-time traffic is statistically guaranteed and the maximum channel utilization is closely approached.

5. CONCLUSION

As a continuation of our previous work [21], in this paper we have proposed a simple and effective call admission control and rate control scheme (CARC) to support QoS of real-time and streaming traffic in the 802.11 wireless LAN. Based on the novel use of the channel busyness ratio, which is shown to be able to characterize the network status, the scheme enables the network to work at the optimal point. Consequently, it statistically guarantees stringent QoS requirements of real-time services, while approaching the maximum channel utilization.

Furthermore, the rate control scheme for ad hoc mode has its own virtue. It provide another kind of distributed solution, i.e., rate control over the packets in outgoing queue without modification to the medium access mechanism in the IEEE 802.11 DCF protocol, to maximize the



(a) (b) (c)

Figure 3. Ad hoc mode: the number of real-time and TCP flows increases over time.

network throughput, and has stable performance under different number of active nodes and different packet size in the presence of all the CBR, VBR and TCP traffic.

Combining the analytical results in our previous work [21] and our proposed CARC scheme, we therefore make it clear that the IEEE 802.11 WLAN can provide statistical QoS guarantees, not just differentiated service, for multimedia services.

References

[1] I. Ada and C. Castelluccia. Differentiation mechanisms for IEEE 802.11. *IEEE INFOCOM'01*, Anchorage, Alaska, April 2001.

[2] Y. Bejerano and R. Bhatia. MiFi: A Framework for Fairness and QoS Assurance in Current IEEE 802.11 Networks with Multiple Access Points. In *Proc. IEEE INFOCOM'04*, 2004.

[3] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. RFC 2475 An architecture for differentiated services. Internet Engineering Task Force, 1998.

[4] D. Clark, S. Shenker, and L. Zhang. Supporting real-time application in an integrated services packet network: architecture and mechanism. in *Proc. of ACM SIGCOMM*, 1992.

[5] H. Garcia-Molina. Elections in a distributed computing system. *IEEE Trans. Comp.*, vol. 31, no. 1, Jan. 1982.

[6] *IEEE standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*, ISO/IEC 8802-11: 1999(E), 1999.

[7] *Draft Supplement to Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications: Medium Access Control (MAC) Enhancements for Quality of Service (QoS)*, IEEE Std 802.11e/D8.0, Feb. 2004.

[8] ITU-T G.114. One-way transmission time, 1996.

[9] ITU-T G.1010. End-user multimedia QoS categories, 2001.

[10] S. Jamin, P.B. Danzig, S. Shenker, and L. Zhang. A measurement-based admission control algorithm for integrated service packet networks. *IEEE/ACM Transactions on Networking*, vol. 5, no. 1, Feb. 1997.

[11] A. Lindgren, A. Almquist, and O. Schelen. Evaluation of quality of service schemes for IEEE 802.11 wireless LANs. In *Proc. Local Computer Networks*, 2001.

[12] S.C. Lo, G. Lee, and W.T. Chen. An Efficient Multipolling Mechanism for IEEE 802.11 Wireless LANs. *IEEE Trans. Computers*, 52(6): 764-778 (2003)

[13] The network simulator ns-2. http://www.isi.edu/nsnam/ns.

[14] H. Perros and K. Elsayed. Call admission control schemes: a review. *IEEE Communications Magazine*, Nov. 1996.

[15] S.-T. Sheu and T.-F. Sheu. A bandwidth allocation/sharing/extension protocol for multimedia over IEEE 802.11 ad hoc wireless LANs. *IEEE Journal on Selected Area in Communications*, Oct. 2001.

[16] S. Singh and J. Kurose. Electing 'good' leaders. *J. Par. Distr. Comput.*, vol. 18, no. 1, May 1993.

[17] J. L. Sobrinho and A. S. Krishnakumar. Real-time traffic over the IEEE 802.11 medium access control layer. *Bell Labs Tech. J.*, Autumn 1996.

[18] A. Veres, A. T. Campbell, M. Barry, and L.-H. Sun. Supporting service differentiation in wireless packet networks using distributed control. *IEEE Journal on Selected Area in Communications*, Oct. 2001.

[19] Y. Xiao, H. Li, and S. Choi. Protection and guarantee for voice and video traffic in IEEE 802.11e Wireless LANs. *IEEE INFOCOM'04*, Hong Kong, China, March 2004.

[20] J.Y. Yeh and C. Chen. Support of multimedia services with the IEEE 802-11 MAC protocol. In *Proc. ICC'02*, 2002

[21] H. Zhai, X. Chen, and Y, Fang, How Well Can the IEEE 802.11 Wireless LAN Support Quality of Service? Submitted for publication. Available at http://www.ecel.ufl.edu/~zhai/publication/qoswlan.pdf

[22] H. Zhai, Y. Kwon, and Y. Fang. Performance Analysis of IEEE 802.11 MAC Protocols in Wireless LANs. Accepted for publication in *Wiley Wireless Communications and Mobile Computing*, 2004.