# Scheduling of real-time traffic in IEEE 802.11 wireless LANs *

Constantine Coutras [a], Sanjay Gupta [b] and Ness B. Shroff [c]

[a] *Department of Computer Science, Illinois Institute of Technology, Chicago, IL 60616, USA*
[b] *Motorola, GSM Products Division, 1501 West Shore Drive (IL27-3223), Arlington Heights, IL 60004, USA*
[c] *School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907, USA*

The desire to provide universal connectivity for mobile computers and communication devices is fueling a growing interest in wireless packet networks. To satisfy the needs of wireless data networking, study group 802.11 was formed under IEEE project 802 to recommend an international standard for *Wireless Local Area Networks* (WLANs). A key part of the standard are the Medium Access Control (MAC) protocols. Given the growing popularity of real-time services and multimedia based applications it is critical that the 802.11 MAC protocols be tailored to meet their requirements. The 802.11 MAC layer protocol provides *asynchronous, time-bounded, and contention free access control* on a variety of physical layers. In this paper we examine the ability of the point coordination function to support time bounded services. We present our proposal to support real-time services within the framework of the point coordination function and discuss the specifics of the connection establishment procedure. We conduct performance evaluation and present numerical results to help understand the issues involved.

## 1. Introduction

In recent years there has been an increasing trend towards personal computers and workstations becoming “portable” and “mobile”. This ever increasing group of mobile users have been demanding access to network services similar to their “tethered” counterparts. The desire to provide universal connectivity for these portable mobile computers and communication devices is fueling a growing interest in wireless packet networks. To meet these and other future communication needs it is expected that tomorrow’s communication networks will employ wireless media in the local area and utilize high capacity wired media in the metropolitan and wide-area environment. Wireless systems and networks will provide communication capability, not only between mobile terminals, but also permit these mobile devices to have access to “wired” networks.

In order to achieve the goal of offering broadband communication services and providing universal connectivity to mobile users it is important that (i) a suitable standard for *Wireless Local Area Networks* (WLANs) be designed and (ii) an approach to interconnect these WLANs to the existing wired LANs and broadband networks be developed. A key design requirement for WLANs is that mobile hosts be able to communicate with other mobile and “wired” hosts (on other IEEE 802 LANs and/or networks) in a transparent manner, i.e.

(i) a WLAN should appear to the *Logic Link Control* (LLC) layer and those above as just another 802.x LAN (for example, Ethernet and Token ring), and

(ii) the response times should not be so large that the productivity of end-users is compromised.

To be able to achieve the above objectives it is imperative that mobility be handled at or below the Media Access Control (MAC) layer (note that in wireless networks an “address” does not correspond to a fixed physical location as in wired networks). Furthermore, it is important that the performance available to mobile users be comparable to the performance available to the wired hosts.

To satisfy the above mentioned needs of wireless data networking, study group 802.11 was formed under IEEE project 802 to recommend an international standard for WLANs. The scope of the 802.11 study group is to develop MAC and physical layer standards for wireless connectivity of fixed, portable, and mobile stations within a local area. Specifically, the 802.11 standard will describe:

(1) functions and services required by an 802.11 compliant device to operate within a wireless network as well as aspects of station mobility within these networks;

(2) MAC procedures to support asynchronous and time-bounded delivery of data frames; and

(3) services required to provide security and privacy to 802.11 compliant devices.

The physical layers to be considered are Direct Sequence Spread Spectrum (DSSS), Frequency Hopping Spread Spectrum (FHSS), and Diffuse Infrared. The operating frequency range that has been allocated for the 802.11 WLANs differs from country to country. For example, in the United States, a frequency range from 2.4 GHz to 2.4835 GHz has been allocated for DSSS.

Given the growing popularity of real-time services and multimedia-based applications it is critical that the 802.11 MAC protocols be tailored to meet their requirements. The 802.11 MAC layer protocol provides *asynchronous, time-bounded, and contention free access control* on a variety of physical layers. A previous study [2,3] examined the

* This research is in part supported by the National Science Foundation through grants NCR 9624525, CDA 9422250, CDA 96-17388, and ANI 98-0544198.

performance of the asynchronous data transfer methods of the IEEE 802.11 WLAN protocol. The point coordination function provided by the 802.11 MAC protocol is designed to support time-bounded services, essential for transporting real-time services such as voice. In this paper we examine the ability of the Point Coordination Function (PCF) of the IEEE 802.11 standard to support time-bounded services and propose appropriate modifications where necessary.

The paper is organized as follows. In section 2 we discuss the 802.11 standard and the MAC protocols provided as a part of the standard. Section 3 presents our proposal to support real-time services within the framework of the point coordination function. In section 4 we discuss the specifics of the connection establishment procedure and present numerical results. Finally, we conclude in section 5 by presenting areas of future research.

## 2. Overview of the IEEE 802.11 draft standard

An 802.11 network, in general, consists of Basic Service Sets (BSS) that are interconnected with a *Distribution System* (DS) as shown in figure 1. Each BSS consists of mobile nodes, henceforth referred to as *stations*, that are controlled by a single *Coordination Function* – the logical function that determines when a station transmits and receives via the wireless medium. Stations in a BSS gain access to the DS and to stations in “remote” BSSs through an *Access Point* (AP). An AP is an entity that implements both the 802.11 and the DS MAC protocols, and can therefore communicate with stations in the BSS to which it belongs, and to other APs (that are connected to the DS). Before a station can access the wireless medium it has to be associated with an AP. A station can be associated with only one AP at any given time. The DS supports mobility by providing services necessary to handle the address[1] to destination mapping and the integration of BSS’s in a manner that is transparent to stations, i.e. hosts (either mobile or wired) do not need to know the physical location of other hosts for communication. A network of interconnected BSSs, in which mobiles can roam without loss in connectivity, is shown in figure 1.

In this setting we now briefly describe the IEEE 802.11 MAC layer protocol.

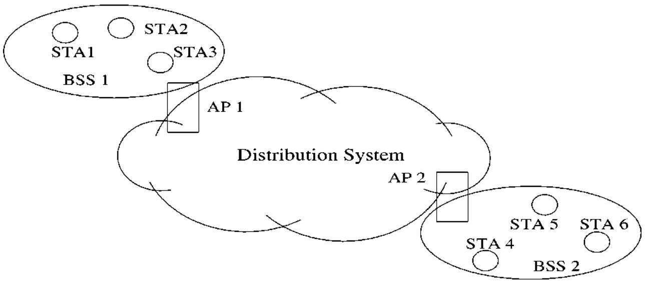


Figure 1. Components of an 802.11 WLAN system.

### 2.1. MAC layer protocol for IEEE 802.11 WLANs

The 802.11 MAC layer protocol provides *asynchronous, time-bounded, and contention free access control* on a variety of physical layers. These functions are provided independently of the characteristics of the underlying physical layers and/or data rates. The basic access method in the 802.11 MAC protocol is the *Distributed Coordination Function* (DCF) which is known as *Carrier Sense Multiple Access with Collision Avoidance* (CSMA/CA). In addition to the DCF, the 802.11 also incorporates an alternative access method known as the *Point Coordination Function* (PCF) – an access method that is similar to “polling” and uses a point coordinator (usually the AP) to determine which station has the right to transmit. The PCF has been developed for providing real-time services. In this paper, since we are interested in examining (and appropriately refining) the ability of the 802.11 draft standard to support real-time services, we focus on the PCF. Before we discuss PCF, however, we first briefly review the DCF.

### 2.2. Distributed coordination function

When using the DCF, a station, before starting its transmission, senses the channel to determine if another station is transmitting. The station proceeds with its transmission if the medium is determined to be idle for an interval that exceeds the *Distributed InterFrame Space* (DIFS). In case the medium is busy the transmission is deferred by the station until the end of the ongoing transmission. A random interval, henceforth referred to as the *backoff interval*, is then selected. The backoff timer is decremented only when the medium is idle; it is frozen when the medium is busy. Decrementing the backoff timer resumes only after the medium has been free longer than DIFS. A station can initiate transmission when the backoff timer reaches zero. To reduce the probability of collisions, after each unsuccessful transmission attempt, the backoff time is increased exponentially until a given maximum is reached.

*Immediate positive acknowledgements* are employed to determine the successive reception of each data frame (explicit acknowledgements are required since a transmitter cannot determine if the data frame was successfully received by listening to its own transmission as is the case in wired networks). This is accomplished by allowing the receiver to transmit an acknowledgement after a time interval *Short InterFrame Space* (SIFS) (that is less than DIFS) immediately following the reception of the data frame. Acknowledgements are transmitted without the receiver sensing the state of the channel. In case an acknowledgement is not received, the data frame is presumed lost and a retransmission of the data frame is scheduled (by the transmitter). This access method, henceforth referred to as *Basic Access*, is summarized in figure 2.

[1] The study group has chosen to use the IEEE 802 48 bit address space for 802.11 WLANs.

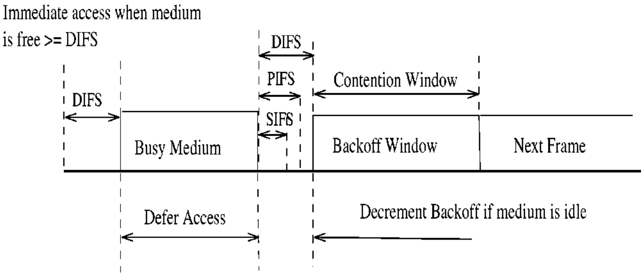


Figure 2. Basic channel access method.

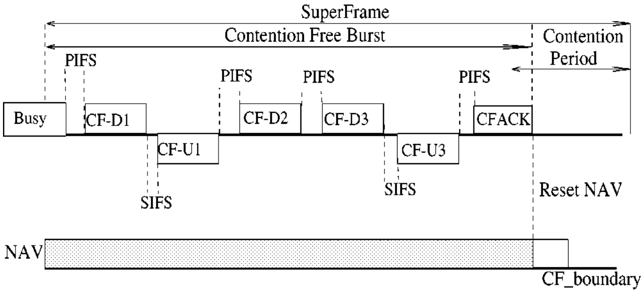


Figure 3. Point coordination function.

### 2.3. Point coordination function

The PCF is built using the DCF through the use of an access priority mechanism that provides synchronous or asynchronous data frames *contention free* access to the channel. In this case, contention and contention free periods alternate with each other as shown in figure 3. A contention free period (during which PCF is active) and the following contention period (during which DCF is active) are together referred to as a *SuperFrame* (SF).

At the beginning of the nominal SF boundary[2] the *Point Coordinator* (PC) senses the channel. If the channel is sensed to be idle, the PC seizes control of the channel by transmitting after it (the channel) has been idle for a time interval *Priority Interframe Space* (PIFS) that is chosen to be smaller than DIFS but larger than SIFS. However, if the medium is determined to be busy, the PC monitors the channel until it is idle, and then seizes its control by transmitting after the channel has been idle for PIFS. The PC maintains control of the channel throughout the contention free period by initiating transmissions after an idle period of PIFS. The transmission of a Contention Free Acknowledgement (CFACK) frame by the PC marks the end of the contention free period.

The PC sends data to stations in CF-Down frames which also achieve the polling function. A "poll bit", if enabled, in the CF-Down frame polls the destination of the CF-Down frame. A station transmits data in CF-Up frames. These CF-Up frames can be transmitted after the reception of a CF-Down frame with the poll bit enabled. The need for separate acknowledgements is avoided by "piggybacking" acknowledgements on subsequent frames (by the setting of an appropriate bit). A station that has no data frame to send following a CF-Down frame (addressed to it) does not transmit a frame and, therefore, the previous CF-Down frame is not acknowledged.

CF-Up frames are transmitted after the channel has been idle for a time interval SIFS as compared to CF-Down frames that are transmitted once the channel has been idle for PIFS. Thus the PC will transmit the next CF-Down frame in case there is no CF-Up frame in time interval PIFS after the transmission of the previous CF-Down frame. To minimize collisions during the contention free periods, each station sets its NAV equal to the maximum allowable length of the CF period. However, a station resets its NAV if a CF-ACK frame is seen by it before its NAV has expired. Refer to figure 3 for more details. Station-to-station transfer of data frames is achieved by addressing the CF-Up frame to a destination station which then generates an acknowledgement following the rules of the basic access method; subsequently the PC seizes control of the channel.

The maximum length of the CF period is chosen such that at least one maximum sized 802.11 data frame, henceforth referred to as a MAC Protocol Data Unit (MPDU), can be transmitted in the contention period. Note that the length of a SF can differ from the nominal SF length[3] due to a phenomenon called *superframe stretching*. Superframe stretching refers to the extension of a superframe beyond its nominal end time due to the ongoing transmission in the contention part of the SF (recall, the PC takes control of the channel only after the channel has been idle for a time interval of PIFS). The phenomenon of superframe stretching can lead to SF lengths being both smaller and larger than the nominal SF length. Note that it is the job of the PC to determine (i) appropriate "nominal" start times for the superframes, (ii) maintain polling lists of stations, and (iii) ensure that performance guarantees are met.

## 3. Support for real-time services

Real-time traffic demands strict performance guarantees from the network. For example, a station wanting to transmit real-time voice to another station would require that the maximum delay for data frames be bounded. Before a certain performance level can be guaranteed to a station, the characteristics of the traffic generated by it have to be known so that appropriate resources (transmission time) can be allocated to it. Therefore, stations that require performance guarantees are required to set up a *connection* with the PC. Henceforth, we use "connection"[4] to refer to a contract between the PC and the station(s) where the PC guarantees the performance desired as long as the station does not violate the declared traffic characteristics.

[2] Nominal SF start times refer to the predetermined time instants at which SF's are supposed to start.

[3] Nominal SF length refers to the time difference between two consecutive SF start times.

[4] Note that the connection could be between the PC and a station or a pair of stations interested in exchanging time bounded data frames and the PC.

Before we proceed with the details of our proposal to support real-time services (under the scope of the PCF), we review the key functionalities provided by the PCF, its most important properties, and the assumptions we make. The PC maintains a polling list that specifies the order in which stations are to be polled. The order in which the stations are polled is dynamic, i.e. it can be changed from one SF to the next. Typically, the polling list contains stations that have established connections with the PC. In addition, the PC *may* decide to poll stations that have not established connections. For simplicity we make the following assumptions:

- The nominal length and start times of each superframe (and, therefore, the nominal start times of the CF periods) is predetermined.
- The effect of hidden stations on the operation of the PCF is ignored.
- The contract associated with a connection cannot be renegotiated at any time.[5]

In order to provide support for real-time services it is essential to determine the time instants when any given station that has established a connection with the PC is to be polled. This critical issue is addressed in the following subsection.

Let $C$ denote the raw transmission rate of the channel. Throughout we use $l_{\text{type}}$ and $t_{\text{type}}$ to denote the length of a "type" frame in bits and time units, respectively. For example, $l_{\text{MPDU}}$ is the length of an MPDU in bits and $t_{\text{MPDU}}$ is the *time* it takes to transmit an MPDU of length $l_{\text{MPDU}}$ bits. For notational simplicity we include the overhead associated with the transfer of various frames in their length. For example, $l_{\text{MPDU}}$ includes the overhead due to the CF-Down frames, the idle time (SIFS) a station has to wait to transfer a CF-Up frame following a CF-Down frame, and the time interval PIFS which the PC must wait before initiating the next CF-Down frame. We let $t(n)$, $n = 1, 2, \ldots,$ denote the nominal start time of the $n$th superframe and let $t'(n)$ denote the actual start time of the $n$th superframe. Further, $t_{\text{SF}}$ is defined to be the nominal length of a superframe ($t_{\text{SF}} = t(n+1) - t(n)$) and $t_{\text{SF}}(n)$ and $t_{\text{CF}}(n)$ denote the length of the $n$th superframe and contention free interval, respectively.

A polling list consists of stations transmitting synchronous and asynchronous data frames.[6] Stations transporting synchronous frames (with existing connections) should be given priority over stations transferring asynchronous data frames since no guarantees are provided to them (asynchronous stations). Within the set of synchronous stations the order in which the stations are polled should be such that the performance requirements of each are met even in the presence of both arriving and departing connection requests, and statistical variations in the amount of data transferred by various stations. Throughout this paper we assume that the performance requirements are expressed in the form of a 2-tuple $(D, \varepsilon)$. A performance requirement of $(D, \varepsilon)$ indicates that no more than $\varepsilon$ fraction of the traffic should exceed a delay of $D$ time units.

To achieve the above performance objectives we propose that the PC be provided with the ability to guarantee each connection request it accepts "time window(s)" in each superframe within which it will be polled, and thus allowed to *initiate* the transfer of a data frame. In order to provide strict delay and loss guarantees it is important that (i) the position of this time window (relative to the nominal starting time of the SF) not change from one superframe to the next, and (ii) the length of the time window be as small as possible. Intuitively, minimizing the tail distribution of the difference between nominal and actual poll times of any given station will lead to smaller delays (or smaller tail delay distributions). This is important since smaller tail delay distributions leads to more efficiency, especially for real-time traffic. To meet the requirements of real-time traffic we *propose* that

(P1) Each station be polled only once during each superframe.

(P2) Following a CF-Down frame each synchronous station (one that has established a connection) will send a single frame. The maximum frame size that can be used by a station should be negotiated at the time of connection establishment. (Recall that following a CF-Down frame (poll) each station can transmit no more than a single maximum sized MPDU.)

(P3) An arriving connection request, if accepted, is placed at the end of the polling list.

(P4) Following the departure of a connection request, the time window for each station in the polling list that is polled after the departing connection request, is advanced by the time allocated to the departing connection request in each superframe.

Before we discuss our proposal, we would like to emphasize that (P1) and (P2) above do not reflect a significant departure from the 802.11 draft standard. The case where the negotiated maximum frame size exceeds the maximum allowable data frame size can be thought of as one where the station under question is polled more than once in each CF period. Further, since valuable transmission time is wasted in polling, it is not desirable to poll a station more than once in each CF period.

**Fact 1.** A window size of $t_{\text{MPDU}}$ is the smallest that can be guaranteed.

*Proof.* Recall that the difference between the nominal and the actual starting time of a SF can be as large as $t_{\text{MPDU}}$. Therefore, the fact is trivially true. □

[5] A contract renegotiation can be treated as a simultaneous termination and initiation of a connection request and, therefore, can be included in the framework of the current study.

[6] The PC monitors the transmissions during the DCF and will frequently poll stations (time permitting) whose transmission attempts have failed repeatedly even when they have not established connections.

We will now show that a window size of $t_{\mathrm{MPDU}}$ can in fact be *achieved* by our proposal. We first make the following observations.

1. The PC can advance the position of the time window for any given station in a SF if it promises to maintain the new position (with an acceptable variance) in all of the successive superframes. Therefore, the cumulative transmission time available to any station, after the service time window has been advanced, is at least as large as would have been made available to it had the time window not been advanced.

2. For any given station $A$, the set of stations that are polled after it, in any given SF, have no effect on the instant at which $A$ initiates transmission to outside the service time window, in the SF under consideration.

Now we proceed to show that a window of size $t_{\mathrm{MPDU}}$ can be achieved by our proposal. The time at which a connection is polled in any given SF depends on (i) the start time of the SF, (ii) the size of the data frames transferred by stations polled before the connection of interest, and (iii) the arrival and departure of new connections to the polling list. We address each of the above individually:

1. *Statistical variations*. Consider any station $i$ and any given SF; if all stations polled before station $i$ in the SF fully utilize their allocated transmission times, the starting service time for station $i$ relative to the starting time of the superframe does not change and, therefore, the polling time for station $i$ lies within its preassigned window. However, if some of the stations that were polled before station $i$ do not use the entire service time allocated to them, station $i$ could be polled before its preassigned time window. *In this case we propose that the PC poll an asynchronous station until the next polling instant lies in the service time window of station $i$.* Since an asynchronous station is allowed to transmit for no more than $t_{\mathrm{MPDU}}$ each time it is polled, a service time window of length $t_{\mathrm{MPDU}}$ suffices. If no asynchronous station is ready to transmit then station $i$ can be polled twice.

2. *Arriving connection*. An arriving station, if it can be accepted (this issue will be explored in detail subsequently), is put at the end of the polling list (recall we assume no asynchronous stations exist in the polling list). Thus its addition does not impact the position of the service time window for any existing connection.

3. *Departing connection*. Consider the impact of station $i$ departing from the system (i.e. closing its connection). In this case the relative position of the service time window for all stations that follow station $i$ in the polling list is *decremented* by the service time for station $i$ in each SF. From our earlier discussion it follows that this has little impact on the service received by all existing connections in the polling list.

Under the above proposal, we next study how both continuous bit rate and variable bit rate services should be handled by the PC.

## 4. Connection establishment

Before a connection can be established for a given station it has to be ensured that the required transmission time can indeed be made available to the requesting station in the CF period of the superframe. Recall that the maximum allowable length of the CF period is such that at least one maximum sized data frame can be transmitted during the following contention period. However, $\mathrm{CF}^* := \mathrm{SF} - 2\,\mathrm{MPDU}$ is the maximum CF length that can be used to guarantee transmission time to stations transmitting synchronous data frames. Therefore, the sum of the allowable transmission times of all (synchronous) stations in the polling list should not exceed $\mathrm{CF}^*$. Observe that the PC *may* choose a maximum contention-free interval length that is smaller than $\mathrm{CF}^*$ in order to allow more time for the transfer of asynchronous data frames during the contention period.

We now outline procedures to determine the service time required in each SF to meet the demands of a connection request. Before we begin, it is instructive to look at the availability of the channel from the perspective of a station that is allowed to transmit $l^*$ bits each time it is polled (see figure 4). Figure 5 shows the upper and lower bounds and the time actually used by the first station in the polling list.

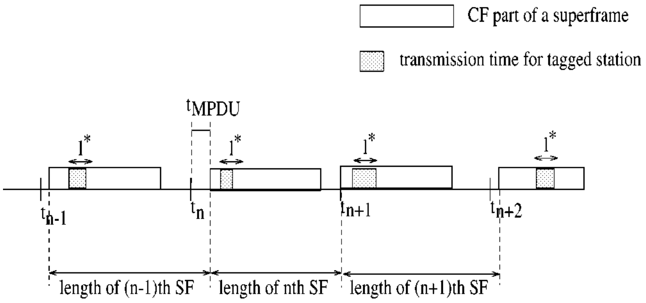


Figure 4. Support for synchronous traffic.

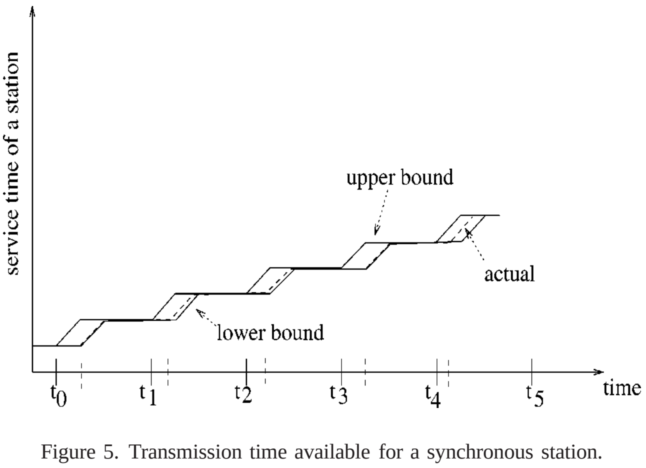


Figure 5. Transmission time available for a synchronous station.

We begin outlining how continuous bit rate services should be handled by the PC. The case of variable bit rate services is considered subsequently.

### 4.1. Continuous bit rate services

Consider a continuous bit rate source with rate $r$ bits per second and a maximum delay requirement of $D$ seconds for all data frames, i.e. a performance requirement of $(D,0)$. Let $l_r^*(D)$ denote the minimum amount of service needed by a CBR source with rate $r$ in each superframe in order to meet the delay bound $D$. Since the time between the start of the CF period of a SF and the end of the CF period of the previous SF could be as large as $2t_{\text{MPDU}}$, the minimum delay that can be *guaranteed* to any connection is limited by $2t_{\text{MPDU}}$. However, given that we propose to poll a station no more than once in each superframe and the difference between the actual and the nominal start times of a SF could be as large as $t_{\text{MPDU}}$; the minimum delay that can be guaranteed is $t_{\text{SF}}+t_{\text{MPDU}}$. Henceforth, throughout the remaining part of the discussion we assume that $D \geqslant t_{\text{SF}}+t_{\text{MPDU}}$.

Assume that the station buffer is empty at the time instant a connection is established. Consider a connection that generates data at rate $r$ and requires a maximum delay that is less than $D$. Consider that this connection is allowed to transfer in each superframe data that is equal to that it generates in $t_{\text{SF}}$, the length of a nominal superframe, i.e. $l_r^*(D)=rt_{\text{SF}}$. It is easy to show using a sample path argument that the worst case delay is less than $D$, the worst case scenario occurs when the first time a station is polled is $t_{\text{SF}}+t_{\text{MPDU}}$ after the connection was established. Observe that since $rt_{\text{SF}}$ is the minimum transmission time required for a station in each SF, no other transmission time allocation policy can do better. Therefore, the transmission resources that must be allocated to a CBR source with rate $r$ and maximum delay requirement $D$ is simply $l_r^*(D)=rt_{\text{SF}}$.

### 4.2. Variable bit rate services

We now estimate the service time required in each SF for variable bit rate (VBR) sources. From the point of view of a single station the channel can be modeled as one that alternates between two states – “Off” and “On” in which it has a capacity of 0 and $C$, respectively. Under the proposals put forth in this paper the maximum time spent in the “On” and “Off” states is fixed (note that the exact time spent in each state is a random variable). For analytical tractability, we characterize the channel by an $N$-state Markov Modulated Fluid (MMF) source. Let $C_n$ denote the channel capacity when the channel is in state $n$, $n=1,\dots,N$. Let $\beta_{ij}, i\neq j, i,j=1,\dots,N$, be the rate at which the channel moves from state $i$ to state $j$; further we define $\beta_{ii} := -\sum_{j=1,j\neq i}^{N}\beta_{ij}$. In section 4.2.1 we discuss the specific choice of parameters (the number of states and the transition rates between states) that enable us to model the channel with sufficient accuracy. First, however, we consider how to determine the buffer occupancy and delay distributions when both the arrival and service processes are modeled by general Markov modulated fluid sources.

Consider a station that generates traffic according to an $M$ state MMF source. The arrival process is modeled by an MMF source since many real-time traffic sources, such as voice and video, have been characterized by Markov modulated fluid processes in the literature [4–8]. Let $\lambda_m$ be the rate at which traffic is generated when the station is in state $m, m=1,\dots,M$, and let $\alpha_{ij}, i\neq j, i,j=1,\dots,M$, be the rate at which the station moves from state $i$ to state $j$; further we define $\alpha_{ii}=-\sum_{j=1,j\neq i}^{M}\alpha_{ij}$.

Let $P_{mn}(t,x)$ denote the probability that at time $t$:

(i) the station is in state $m$,

(ii) the channel is in state $n$, and

(iii) the station buffer contents are $\leqslant x$.

Generalizing the approach first developed in [1], we have

$$\begin{aligned}&P_{mn}(t+\Delta t,x)\\&\quad=\sum_{i=1,i\neq m}^{M}\alpha_{im}\Delta tP_{in}(t,x)+\sum_{i=1,i\neq n}^{N}\beta_{in}\Delta tP_{mi}(t,x)\\&\qquad+\big[1+(\alpha_{mm}+\beta_{nn})\Delta t\big]P_{mn}\big(t,x-(\lambda_m-C_n)\Delta t\big)\\&\qquad+\mathrm{o}(\Delta t),\quad m=1,\dots,M;\ n=1,\dots,N.\end{aligned}\tag{4.1}$$

Here, note that a function $f$ is said to be $\mathrm{o}(h)$, if $\lim_{h\to0}(f(h)/h)=0$. Therefore, dividing both sides by $\Delta t$ and taking the limit as $\Delta t\to0$ in (4.1) we have

$$\begin{aligned}&\frac{\partial P_{mn}}{\partial t}+(\lambda_m-C_n)\frac{\partial P_{mn}}{\partial x}\\&\quad=\sum_{i=1}^{M}\alpha_{im}P_{in}(t,x)+\sum_{i=1}^{N}\beta_{in}P_{mi}(t,x).\end{aligned}\tag{4.2}$$

The steady state behavior of the system can be obtained by setting $\partial P_{mn}/\partial t=0$ in (4.2). Let $F_{mn}(x)$ denote the buffer occupancy distribution in steady state, i.e. $F_{mn}(x)=\lim_{t\to\infty}P_{mn}(t,x)$. It then follows from (4.2) that

$$(\lambda_m-C_n)\frac{\partial F_{mn}}{\partial x}=\sum_{i=1}^{M}\alpha_{im}F_{in}(x)+\sum_{i=1}^{N}\beta_{in}F_{mi}(x).\tag{4.3}$$

Denote by $\pi_m^{\mathrm{s}}, m=1,\dots,M$, the steady state probability of the station being in state $m$. Similarly, let $\pi_n^{\mathrm{c}}, n=1,\dots,N$, be the probability of the channel being in state $n$ in steady state. From the definition of $F_{mn}(\cdot)$ it follows that

$$F_{mn}(\infty)=\pi_m^{\mathrm{s}}\pi_n^{\mathrm{c}},\quad m=1,\dots,M;\ n=1,\dots,N.\tag{4.4}$$

The set of equations given by (4.4) represents the boundary conditions required to solve the set of differential equations in (4.3).

Let $\mathbf{F}$ be an $(MN \times 1)$ vector of $F_{mn}, m = 1, \dots, M$; $n = 1, \dots, N$, arranged in lexicographic order, i.e. $\mathbf{F} = (F_{11}, \dots, F_{1N}, F_{21}, \dots, F_{MN})^{\mathrm{T}}$, and let $\boldsymbol{\Lambda}$ be an $MN \times MN$ diagonal matrix with $\lambda_m - C_n$ as the $[(m-1)N+n]$th diagonal element. Further, let $\boldsymbol{\Theta}$ be an $MN \times MN$ matrix of elements

$$\boldsymbol{\Theta}_{j,k} = \begin{cases} \alpha_{m_1m_1} + \beta_{n_1n_1} & \text{if } m_1 = m_2, n_1 = n_2, \\ \alpha_{m_2m_1} & \text{if } m_1 \neq m_2, n_1 = n_2, \\ \beta_{n_2n_1} & \text{if } m_1 = m_2, n_1 \neq n_2, \\ 0 & \text{otherwise,} \end{cases}$$

where $j = (m_1 - 1)N + n_1$ and $k = (m_2 - 1)N + n_2$.

The set of differential equations (4.3) can now be expressed as

$$\boldsymbol{\Lambda}\mathbf{F}' = \boldsymbol{\Theta}\mathbf{F}, \tag{4.5}$$

where $\mathbf{F}'$ is an $(MN \times 1)$ vector with the $[(m-1)N+n]$th element as $\partial F_{mn}/\partial x$. For simplicity, assume that $\boldsymbol{\Lambda}$ is invertible[7]; in this case the solution to the set of differential equations (4.3) can be expressed as

$$\mathbf{F} = \sum_{z_i \leqslant 0} a_i \boldsymbol{\psi}_i \mathrm{e}^{z_i x}, \tag{4.6}$$

where $z_i$ and $\psi_i$ are eigenvalue–eigenvector pairs with associated $\boldsymbol{\Lambda}^{-1}\boldsymbol{\Theta}$ and $a_i$ are the constants that need to be determined through the use of boundary conditions. Further, since 0 is an eigenvalue we have

$$\mathbf{F} = \mathbf{F}(\infty) + \sum_{z_i < 0} a_i \boldsymbol{\psi}_i \mathrm{e}^{z_i x}. \tag{4.7}$$

Clearly, boundary conditions in addition to (4.4) are needed to obtain the scalar constants $a_i$. The required boundary conditions are

$$F_{mn}(0) = 0 \quad \text{if } \lambda_m > C_n. \tag{4.8}$$

The above follows from the fact that if the rate of arrival from the source is greater than the service rate of the channel, the probability of the buffer being empty is zero. From the vector $\mathbf{F}$, we can determine the buffer occupancy distribution for our system.

We now determine the distribution of the time spent in the system by a "fluid particle". Let $D_l(x)$ be a random variable that denotes the time spent in the system by a fluid particle that arrives when the channel is in state $l$ and the amount of "fluid" in the buffer is $x$:

$$\begin{aligned} &P\bigl\{D_l(x) \geqslant t\bigr\} \\ &\quad = \sum_{k \neq l} P\bigl\{D_k(x - C_l\Delta t) \geqslant t - \Delta t\bigr\}\beta_{lk}\Delta t \\ &\qquad + P\bigl\{D_l(x - C_l\Delta t) \geqslant t - \Delta t\bigr\}\Bigl[1 - \sum_{k \neq l}\beta_{lk}\Delta t\Bigr]. \end{aligned}$$

It then follows from the above that

$$\begin{aligned} &\frac{P\{D_l(x) \geqslant t\} - P\{D_l(x - C_l\Delta t) \geqslant t - \Delta t\}}{\Delta t} \\ &\quad = \sum_{k \neq l} P\bigl\{D_k(x - C_l\Delta t) \geqslant t - \Delta t\bigr\}\beta_{lk} \\ &\qquad - P\bigl\{D_l(x - C_l\Delta t) \geqslant t - \Delta t\bigr\}\sum_{k \neq l}\beta_{lk}. \end{aligned}$$

Defining $\mathcal{D}_l(t,x) = P\{D_l(x) \geqslant t\}$ it follows that

$$\frac{\partial \mathcal{D}_l(t,x)}{\partial t} + C_l\frac{\partial \mathcal{D}_l(t,x)}{\partial x} = \sum_{k \neq l}\bigl[\mathcal{D}_k(t,x) - \mathcal{D}_l(t,x)\bigr]\beta_{lk}. \tag{4.9}$$

The boundary conditions needed to solve the set of differential equations in (4.9) are

$$\begin{aligned} &\mathcal{D}_l(t,0) = \begin{cases} 1 & \text{if } C_l \geqslant 0 \text{ and } t = 0, \\ 0 & \text{otherwise,} \end{cases} \\ &\mathcal{D}_l(x,x) = 1, \quad x \geqslant 0, \quad l = 1, 2, \dots, N, \end{aligned}$$

$\delta(t) = 1$ if $t = 0$ and is 0 otherwise.

Now let $D(x)$ be a random variable that denotes the time spent in the system by a fluid particle that arrives when the amount of fluid in the buffer is $x$. It follows from a simple conditioning argument that

$$P\bigl\{D(x) \geqslant t\bigr\} = \sum_{n=1}^{N}\sum_{m=1}^{M} P\bigl\{D_n(x) \geqslant t\bigr\}\,\mathrm{d}F_{mn}(x) \tag{4.10}$$

and the probability that the time spent (a random variable) in the system by a fluid particle exceeds $t$

$$\begin{aligned} P\{D \geqslant t\} &= \int_0^{\infty}\sum_{n=1}^{N}\sum_{m=1}^{M} P\bigl\{D_n(x) \geqslant t\bigr\}\,\mathrm{d}F_{mn}(x) \\ &= \int_0^{T}\sum_{n=1}^{N}\sum_{m=1}^{M} P\bigl\{D_n(x) \geqslant t\bigr\}\,\mathrm{d}F_{mn}(x) \\ &\quad + 1 - \sum_{n=1}^{N}\sum_{m=1}^{M} F_{mn}(T) \\ &= 1 - \Biggl(\sum_{n=1}^{N}\sum_{m=1}^{M} F_{mn}(T) \\ &\quad - \int_0^{T}\sum_{n=1}^{N}\sum_{m=1}^{M} P\bigl\{D_n(x) \geqslant t\bigr\}\,\mathrm{d}F_{mn}(x)\Biggr) \\ &= 1 - \sum_{n=1}^{N}\sum_{m=1}^{M}\Biggl(F_{mn}(T) \\ &\quad - \int_0^{T} P\bigl\{D_n(x) \geqslant t\bigr\}\,\mathrm{d}F_{mn}(x)\Biggr). \end{aligned} \tag{4.11}$$

### 4.2.1. Choice of channel model parameters

Consider an $N$ state MMF source and let $\beta_{ij}$, $i \neq j$, $i, j = 1, \dots, N$, be the rate at which a transition from state $i$

[7] $\boldsymbol{\Lambda}$ is not invertible if for any $m, m = 1, \dots, M$, and $n, n = 1, \dots, N$, $\lambda_m = C_n$; in this case the appropriate row and column can be eliminated and the process repeated.

to state $j$ takes place; we assume that $\beta_{ij} = 0$ if $j \neq \lfloor (i+1)/N \rfloor$ and define $p_{ij} = \beta_{ij} / \sum_{k=1}^{N} \beta_{ik} = 1$.

For the case at hand any single station is either provided the entire channel capacity or none at all we choose

$$C_n = \begin{cases} C & \text{if } n \leqslant NS, \\ 0 & \text{otherwise,} \end{cases} \tag{4.12}$$

where *NS* is a chosen state.

Define $X(\beta)$ to be an exponential random variable with rate $\beta$. We suggest choosing $\beta_{i-1,i} = \beta_{i,i+1}$, $i = 2, \dots, NS$. In this case the probability that a station views the channel in the "On" state for time $t$ is

$$P\{T_{\text{on}} = t\} = \beta_{12} \mathrm{e}^{-\beta_{12} t} \frac{(\beta_{12} t)^{NS-1}}{(NS-1)!} \tag{4.13}$$

and the probability that the time spent in the "Off" state is $t$ is

$$P\{T_{\text{off}} = t\} = \beta_{N1} \mathrm{e}^{-\beta_{N1} t} \frac{(\beta_{N1} t)^{(N-NS)-1}}{((N-NS)-1)!}. \tag{4.14}$$

Note that the time the channel spends in the "On" and "Off" states are random variables that are a sum of exponential random variables and by carefully choosing the number of states and transition rates the service time available to a connection can be accurately modeled.

### *4.2.2. Numerical results*

The numerical results are now presented. We are interested in the probability of overflow for various maximum delay times. This will determine the system's performance. In order to conduct performance evaluation of our proposal, two programs were developed, a simulation program, and a program for calculating the theoretical data. Since the analysis we provide is an exact analysis, the results from the numerous test cases were so close (virtually identical) that we do not explicitly show the simulation results.

We now present the case that examines the suitability of our proposal to support real-time voice. Consider a 32 Kbps voice source that has access to a 1 Mbps channel, which polls the source according to our proposal. The source has an average "On" time of 40 ms and an average "Off" time again of 40 ms. The source is modeled by a two-state (one "On" and one "Off" state) MMF source. The channel has a superframe ($t_{\text{SF}}$) of 40 ms. It is modeled by a 16-state (8 "On" and 8 "Off") MMF source. This is done because the channel is much more "deterministic" in its behavior (however, the existence of "superframe stretching" does not allow us to use a deterministic model). The "On" and "Off" times ($T_{\text{on}}$ and $T_{\text{off}}$) for the channel are calculated so that the source gets the appropriate bandwidth. When the bandwidth allocation factor $BA_{\text{factor}}$ is 1.0 the source is given 32 kbits of bandwidth (this, of course, is twice its mean value, since the "On" and "Off" periods are equal).

These are the default values. The graphs that vary some of these values state the values of that variation.

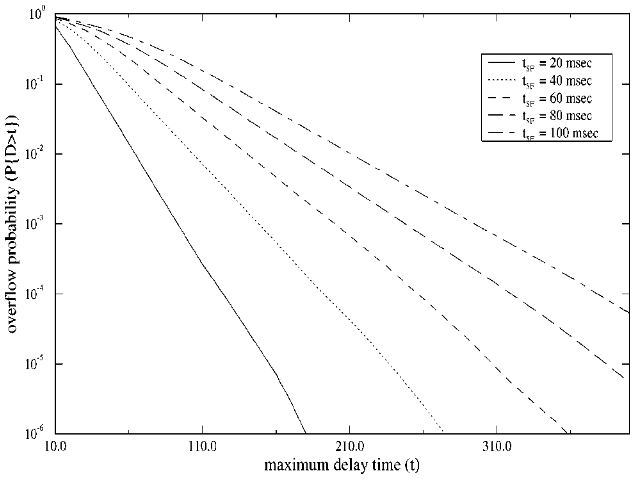


Figure 6. Superframe variation.

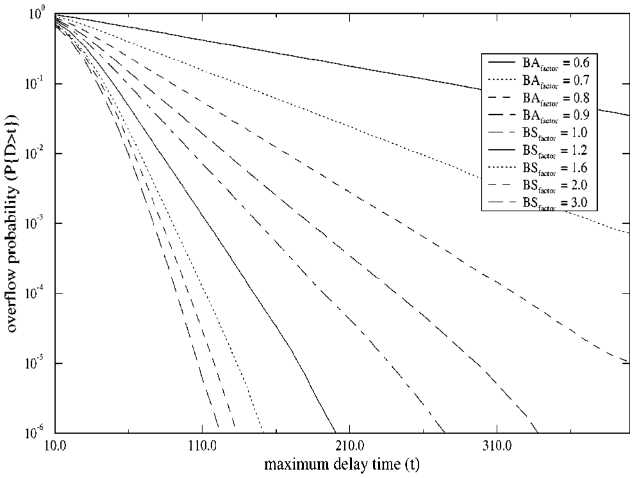


Figure 7. Bandwidth variation.

In figure 6, we show the probability of overflow for various maximum delay time values. Each curve represents a different channel superframe value. The smaller the superframe size $t_{\text{SF}}$, the better the performance. A smaller superframe size actually means smaller intervals between polling times for the source. As we have observed, the superframe size $t_{\text{SF}}$ is a very important parameter in determining the performance of the system. Hence, this implies that the network designer should choose as small a value of $t_{\text{SF}}$ as possible as long as the polling overhead is still negligible.

In figure 7, we show the probability of overflow again, but now each curve is obtained by varying the $BA_{\text{factor}}$. Giving the source a $BA_{\text{factor}}$ that is more than one is actually giving it more than 32 kbps during each superframe. The performance improves as the $BA_{\text{factor}}$ increases, although beyond a $BA_{\text{factor}}$ of two, one observes diminishing returns for this case. Giving the source a $BA_{\text{factor}}$ that is less than one is actually giving it less than 32 kbps during each superframe. As the $BA_{\text{factor}}$ goes towards 0.5 (the bandwidth allocation equal to the mean value of sources rate) the performance drops quickly.

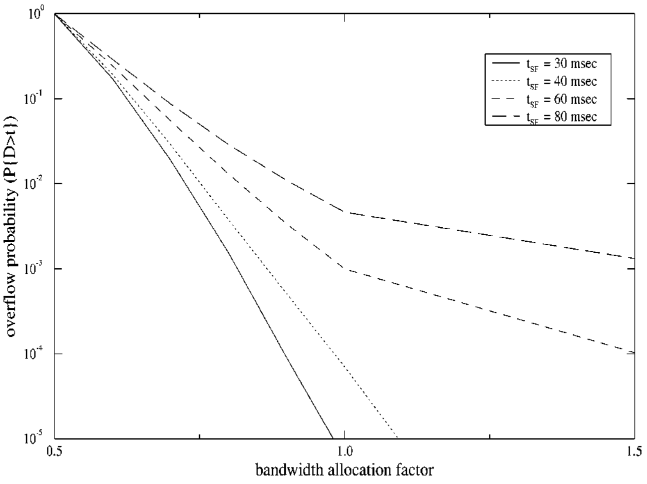


Figure 8. Bandwidth variation for a specific delay (200 ms).

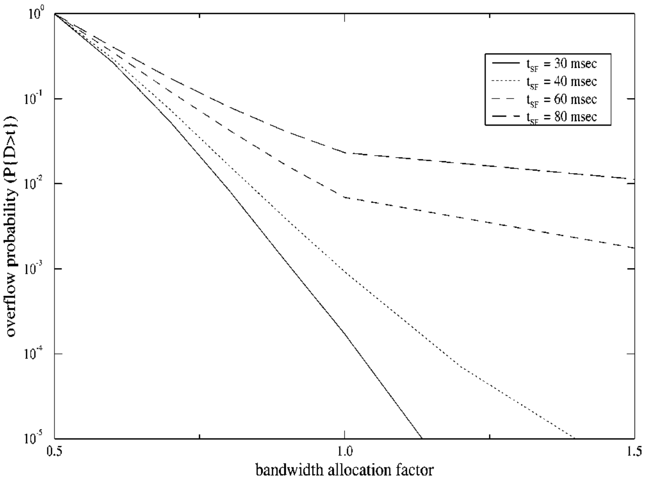


Figure 9. Bandwidth variation for a specific delay (150 ms).

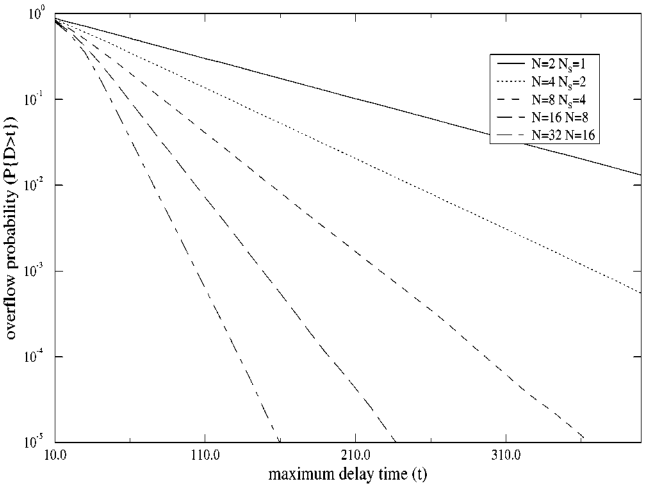
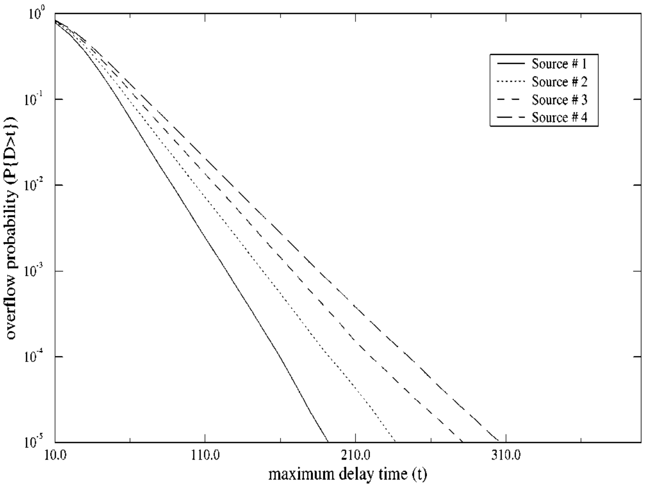


Figure 10. Channel states variation.



| Source | Time “On” | Time “Off” |
|---|---|---|
| Source # 1 | 20 ms | 20 ms |
| Source # 2 | 40 ms | 40 ms |
| Source # 3 | 60 ms | 60 ms |
| Source # 4 | 80 ms | 80 ms |

Figure 11. Source variation.

In figures 8 and 9 we observe the overflow probability for various $BA_{\text{factor}}$ values. Each curve represents a different channel superframe value. The maximum delay time is 200 ms in the first figure and 150 in the second. Again we see the effect (even more clearly now) of the $BA_{\text{factor}}$ on performance. Since each curve represents various superframe sizes $t_{\text{SF}}$ we again observe the significant effect of the superframe size $t_{\text{SF}}$ on overflow probability and the combined effect of superframe size and $BA_{\text{factor}}$.

From the graphs it can be interpreted that if the superframe length is 40 ms and a delay of 200 ms is tolerable, then a $BA_{\text{factor}}$ of 1.0 is sufficient to keep the overflow probability under $10^{-4}$, while it takes a $BA_{\text{factor}}$ of approximately 1.2 for the same overflow probability limit if the tolerated delay is 150 ms. We also observe that for a specific $BA_{\text{factor}}$ we can reduce the superframe length $t_{\text{SF}}$ in order to meet specific requirements of tolerable delay and overflow probability, but again the network designer should choose as small a value of $t_{\text{SF}}$ as possible as long as the polling overhead is still negligible. If a delay of 200 ms is tolerable and an overflow probability of $10^{-4}$ is an upper limit, then for a superframe size $t_{\text{SF}}$ of 40 ms and a $BA_{\text{factor}}$ of 1.0 the channel can support up to 31 voice channels.

In figure 10 we show the probability of overflow for various maximum delay time values. Each curve represents a different number of states that model the channel. Using more states to represent the channel makes its behavior more “deterministic” and its superframe more “stable”. So here we see the effect of the channel’s superframe “stability” on overflow probability. The more the channel states modeling the channel, the less the effect of the superframe stretching, and the better the performance.

Finally, in figure 11 we show the probability of overflow for various maximum delay time values again. Now each curve represents a source with a different pair of “On” and “Off” times. As we can see the smaller the “On” and “Off” periods the better the performance, as the source is less bursty.

## 5. Conclusion

We have presented a brief outline of the IEEE 802.11 MAC protocol and discussed in detail the procedures required for supporting real time traffic. We have then identified the key engineering decisions that must be made in order to efficiently support real time services. Following this, we present an approach that outlines how these decisions should be made in order to most efficiently support real time applications. Under the framework developed, we provide techniques for resource allocation and call admission/rejection in IEEE 802.11 WLANs. We argue that the proposed approach is efficient and suitable for real-time computation.

## References

[1] D. Anick, D. Mitra and M. Sondhi, Stochastic theory of a data handling system with multiple sources, in: *Conf. Rec. 1980 Int. Conf. Commun.*, Seattle, WA (1980).

[2] H. Chhaya, Performance evaluation of the IEEE 802.11 MAC protocol for wireless LANs, Master’s thesis, Department of Electrical Engineering, Illinois Institute of Technology, Chicago (1996).

[3] H.S. Chhaya and S. Gupta, Performance of asynchronous data transfer medhods in IEEE 802.11 MAC protocol, IEEE Personal Communications Magazine 3(5) (1996) 8–15.

[4] J.N. Daigle and J.D. Langford, Models for analysis of packet voice communications systems, IEEE Journal on Selected Areas in Communications 4 (September 1986) 847–855.

[5] A. Elwalid and D. Mitra, Approximations and admission control of a multi-service multiplexing system with priorities, in: *Proc. IEEE INFOCOM ’96*, Boston, MA (April 1995) pp. 463–472.

[6] A.I. Elwalid, Markov modulated rate processes for modeling, analysis and control of communication networks, Ph.D. thesis, Graduate School of Arts and Sciences, Columbia University (1991).

[7] B. Maglaris, D. Anastassiou, P. Sen, G. Karlsson and J.D. Robbins, Performance models of statistical multiplexing in packet video communications, IEEE Transactions on Communications 36(7) (July 1988) 834–844.

[8] N.B. Shroff and M. Schwartz, Improved loss calculations at an ATM multiplexer, in: *Proc. IEEE INFOCOM ’96*, San Francisco, CA (March 1996) pp. 561–568.

**Constantine Coutras** is a Ph.D. candidate in computer science at the Illinois Institute of Technology. He received his M.S. degree in computer science in 1994 from the Rochester Institute of Technology. He also received his Diploma in electrical engineering in 1992 from the University of Patras, Greece. His thesis research includes analytical modeling and performance evaluation of wireless local area networks under asynchronous and real time traffic. His other research interests include broadband networks, fiber optic networks and management and control of networks. He joined Motorola in 1998.
E-mail: coutras@cig.mot.com

**Sanjay Gupta** received his Ph.D. in systems in 1993 and M.S. degree in electrical engineering in 1989, both from the University of Pennsylvania. He received his B.Tech. degree in electrical engineering in 1988 from Indian Institute of Technology, Kanpur. He joined GSM Products Division at Motorola in 1997 where he is working on Universal Mobile Telecommunications System (UMTS). Prior to that he was at the Department of Electrical and Computer Engineering at Illinois Institute of Technology in 1993 as an Assistant Professor. He was a visiting research scholar at Hong Kong University of Science and Technology during the summer of 1995 and 1996. He is currently working on problems in the area of integration of wireless and wireline communication networks. His other research interests include wireless local area networks, ATM based broadband networks, management and control of communication networks, and stochastic modeling and simulation. He is currently editing a special issue of Wireless Networks on error control and has served as a Technical Program Committee member for the IEEE INFOCOM conference since 1993 and the IEEE Vehicular Technology Conference 1995.
E-mail: gupta@cig.mot.com

**Ness B. Shroff** received the B.S. degree from the University of Southern California in 1988, the M.S.E. degree from the University of Pennsylvania in 1990, and the M.Phil. and Ph.D. degrees from Columbia University in 1993 and 1994, respectively. Since 1994, he has been a Faculty member in the School of Electrical and Computer Engineering at Purdue University, where he is currently an Associate Professor. He was the Program Chair for the 1999 IEEE Computer Communications Workshop, and is an Associate Editor for Computer Networks and IEEE Communications Letters. His research interests are in traffic analysis, network control, resource management, and quality of service in high-speed wired and wireless networks. He has authored or co-authored over 45 archival journal and conference publications, and one pending patent. He received the National Science Foundation CAREER Award in 1996.
E-mail: shroff@purdue.edu