Fluid Analysis of Delay and Packet Discard Performance for QoS Support in Wireless Networks

Marwan M. Krunz, *Member, IEEE*, and Jeong Geun Kim, *Student Member, IEEE*

Abstract— **Providing quality of service (QoS) guarantees over wireless links requires thorough understanding and quantification of the interactions among the traffic source, the wireless channel, and the underlying link-layer error control mechanisms. In this paper, we account for such interactions in an analytical model that we use to investigate the delay distribution and the packet discard rate over a wireless link. In contrast to previous studies, our analysis accommodates the inherent autocorrelations in *both* the traffic source as well as the channel error characteristics. An on/off fluid process is used to model the arrival of packets at the transmitter. These packets are temporarily stored in a FIFO buffer before being transmitted over a channel with a time-varying and autocorrelated service rate. Using fluid analysis, we first derive the distribution for the queueing delay at the transmitter. As part of this analysis, we solve a fundamental fluid problem, namely, the probability distribution for the workload generated by a 2-state fluid source over a fixed time interval. We then use the delay analysis to derive the packet discard rate at the receiver (a packet is discarded when the maximum number of retransmissions is reached). A closed-form expression for the effective bandwidth *subject to a delay constraint* is provided as a function of the source, channel, and error scheme parameters. This expression enables fast assessment of the bandwidth requirement of real-time traffic over QoS-based wireless networks. Numerical results and simulations are used to verify the adequacy of the analysis and to study the interactions among various system parameters.**

Index Terms — **Wireless networks, QoS, delay distribution, fluid analysis, packet discard rate.**

I. Introduction

Future broadband wireless technologies are expected to provide a flexible service platform that can support real-time multimedia applications with stringent quality-of-service (QoS) requirements. To provide this support at affordable cost and without disturbing the performance experienced by already established connections, the network must execute a connection admission control (CAC) algorithm, which determines the admissibility of a prospective connection under the given QoS requirements and the available network resources. The CAC decision is made based on predictive knowledge of the expected QoS performance and the required resources for the new connection. Since this decision is to be made *online* during the connection establishment phase, it must rely on *analytical techniques* that quantify the QoS measures of interest in terms of the known system parameters. While the above QoS problem is well known and has been extensively studied in the wireline domain, its wireless counterpart seems to be much more challenging due to the need to explicitly consider the time-varying channel characteristics and the impact of the underlying link-layer error control mechanisms on the network-layer QoS performance.

Two classes of link-level error control are commonly used to improve the performance over the wireless channel: automatic repeat request (ARQ) and forward error correction (FEC). In general, ARQ is used to deliver data requiring high reliability, whereas FEC is more suitable for delay-sensitive traffic. Recent studies (e.g., [4], [15], [32], [23], [29]) suggest that hybrid ARQ/FEC, also known as hybrid ARQ, might be more appropriate for a wireless network that carries traffic with diverse characteristics and QoS requirements. For instance, data connections with relaxed time constraints can use ARQ, while voice and video connections that require low delay, delay jitter, and minimal packet loss may need a combination of FEC and ARQ with time-constrained retransmission [3]. For the sake of generality, we consider in our work a generic hybrid ARQ/FEC technique with separate CRC (cyclic redundancy check) and FEC codes.

Several studies have been conducted on the QoS issue in wireless networks at both the connection and packet levels (see [27], [5], [24], [8], [26], [21], [19], [34], among others). Levine et al. proposed the shadow cluster concept and used it in studying CAC in a wireless network under a guaranteed call dropping probability [21]. In [9] the authors investigated the bandwidth reservation problem under handoff dropping constraints. In [20], [28] the authors advocate a different philosophy to QoS provisioning by means of source-rate adaptation and dynamic bandwidth allocation. Capone and Stavrakakis investigated the region of supportable QoS vectors expressed in terms of the packet dropping probability [5]. Their work, which focused on unbuffered services, provided insight into the resource management aspects for handling diverse QoS constraints. Lu et al. [24] proposed a fair scheduling algorithm for wireless networks that takes into account bursty and location-dependent channel errors. Although their work identified many practical issues, it did not address the interaction between packet scheduling and error control. Throughput

Manuscript received June 10, 1999; revised May 7, 2000, and July 25, 2000. This work was supported by the National Science Foundation under grants ANI-9733143 and CCR-9979310. An abridged version of this paper was presented at the *Int. Conference on Network Protocols (ICNP '99)*, Toronto, Nov. 1999. M. M. Krunz is with the Department of Electrical and Computer Engineering, University of Arizona, Tucson, AZ 85721 (email: krunz@ece.arizona.edu). J. G. Kim is with Qualcomm Inc., San Jose, California (email: jeong@qualcomm.com).

and delay guarantees in the presence of channel contention were provided in [7], [25] but without accounting for channel errors and packet losses during transmission. In [16] we presented a framework for analyzing the QoS performance over a wireless link and used it to investigate the packet loss rate due to buffer overflow at the transmitter. The same framework will be used here to investigate the packet delay distribution and the packet discard rate at the receiver.

Contributions and Paper Organization

In this paper, we analyze the packet delay performance over a wireless link. In contrast to previous studies (e.g., [17], [1], [12]), our analysis accommodates *both* the inherent correlations (i.e., burstiness) in the arriving traffic and the time-varying characteristics of the radio channel. Both aspects are known to have profound impact on the queueing performance. To account for traffic burstiness, we represent the arrival process at the transmitter by a two-state (on-off) fluid process. As for the channel, we capture its time-varying behavior via a two-state (Good/Bad) Markov model, where each state is associated with a given bit error rate (BER). Our channel model is a generalization of the Gilbert-Elliot model [13], [10] for which the two states take their extreme behaviors (i.e., the BERs for the Good and Bad states are zero and 0.5, respectively). We assume a generic hybrid ARQ/FEC scheme with separate error detection and correction codes. Under the above setup, we derive the cumulative distribution function for the packet delay. From a practical standpoint, our analysis can be used to provide *fast* assessment of the QoS performance for *online* CAC, which otherwise cannot be provided by simulation techniques. In practice, one is often interested in determining the required bandwidth subject to given QoS requirements. For that we provide a closed-form expression of the wireless effective bandwidth *subject to a delay constraint*. We use our theoretical results to study the impact of and the interactions among various system parameters and to provide some guidelines on the appropriate selection of these parameters. In arriving at the packet delay distribution, we solve a fundamental problem in fluid-based mathematical modeling, namely the distribution for the workload that is generated by a two-state Markovian source over a fixed interval of time. From the delay distribution, we derive the packet discard rate at the receiver under limited retransmissions. This QoS measure is important for delay-sensitive applications that can tolerate some degree of packet loss but that require prior quantification of this loss to be used, for example, in designing appropriate error concealment mechanisms. By combining the packet discard rate with the buffer overflow rate at the transmitter (which was analyzed in [16]), one can obtain the total packet loss rate over a wireless link.

The rest of the paper is organized as follows. In Section II, we describe the wireless link model. Analysis of the delay performance is provided in Section III. The wireless effective bandwidth is investigated in Section IV. In Section V, the packet discard performance is analyzed. Numerical results and simulations are reported in Section VI, followed by concluding remarks in Section VII.

II. Wireless Link Model

A. Problem Formulation

We consider the wireless link model shown in Figure 1. This model was first used in [16] to study the buffer overflow rate at the transmitter. In here, we use it to evaluate the delay performance and the packet discard rate at the receiver. According to this model, the total channel capacity is shared among several mobile hosts (MHs), each of which is guaranteed a fraction of this total capacity. This is done, for example, by periodic slot assignment in a Time Division Multiple Access (TDMA) system or by using a weighted fair queueing (WFQ) algorithm [24]. In this case, packets generated from or destined to different MHs are often stored in different queues, which for the purposes of this paper can be studied separately. Hence, we focus on the traffic between the base station (BS) and a MH in either direction of the transmission. At the transmitter, incoming packets are temporarily stored in a FIFO queue before being passed to the underlying link layer. The link layer implements a generic hybrid (Type-1) ARQ/FEC error control scheme in which the CRC code is applied first to a packet followed by FEC (i.e., the input to the FEC coder consists of the original packet plus the CRC code). We assume strong CRC code so that the probability of not detecting a packet error is practically zero. In contrast, only a subset of packet errors can be corrected by FEC. In such an ARQ/FEC scheme, the purpose of FEC is to reduce the number of retransmissions, which would possibly improve the delay performance (as demonstrated in this paper). For simplicity, we ignore the protocol overhead of the MAC layer. We also assume, as often the case [12], that the feedback (ACK/NACK) messages are well protected by FEC so that no retransmissions are needed for these messages.

At the receiver, the reverse decoding process is applied. If after CRC decoding a packet error is detected, the receiver sends a negative acknowledgement (NACK) back to the transmitter, triggering a packet retransmission. We impose a limit N_l on the number of packet retransmissions. Once this limit is reached, the packet is discarded at the receiver, irrespective of its error status. We refer to the percentage of discarded packets due to the imposed retransmission limit as the the packet discard rate (PDR). Both the delay performance and the PDR will be obtained as functions of the source, channel, and error control parameters.

B. Queueing Model

We now describe the queueing model that is used to analyze the delay performance over a wireless link. This model was first formulated in [16], where it was used to obtain the queue length distribution and the buffer overflow probability at the transmitter. In this model, the incoming traffic at the transmitter is represented by an on-off fluid process with peak rate r. The on and off periods are exponentially

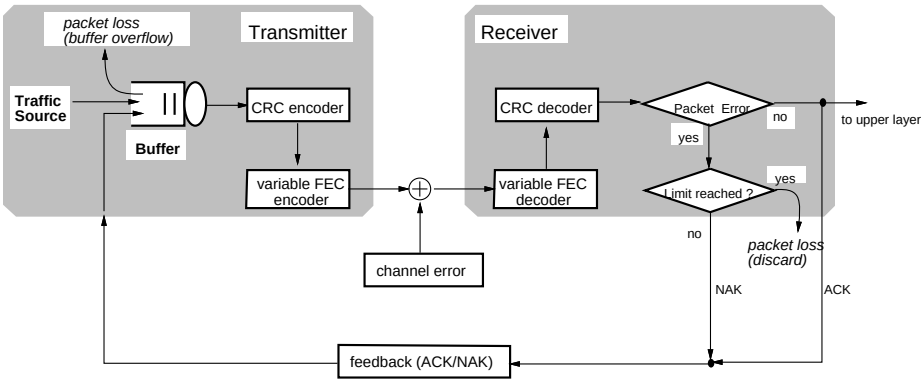


Fig. 1. Wireless link model.

distributed with means $1/\alpha$ and $1/\beta$, respectively. As for the channel, it is represented by a two-state Markov modulated model that alternates between *Good* and *Bad* periods. Such a model has been widely used in the literature as a first-level approximation of slowly varying fading channels [8] (see [33] for a discussion of more general Markov channel models). The BERs during the Good and Bad states are given by P_{eg} and P_{eb}, respectively, where $P_{eg} \ll P_{eb}$. The durations of the Good and Bad states are exponentially distributed with means $1/\delta$ and $1/\gamma$, respectively.

The FEC capability in the underlying hybrid ARQ/FEC mechanism is characterized by the triple (n, k, v), where n is the number of bits in a code block, k is the number of information bits in a code block, and v is the maximum number of correctable bits in a code block. In here, a block corresponds to a fixed-size link-layer packet. Note that n depends on both k and v. The FEC code rate is defined by $e = k/n$. Conditioned on the state of the channel, we assume that bit errors are independent. Accordingly, if the channel was in the Good state during the transmission of a packet, then the probability that this packet will contain an uncorrectable error is given by:

$$P_{c,g} = \sum_{j=v+1}^{n} \binom{n}{j} P_{eg}^{j}(1-P_{eg})^{n-j} \tag{1}$$

A similar expression applies to $P_{c,b}$, the probability of a non-correctable packet error during Bad channel periods, but with P_{eb} replacing P_{eg}.

Following the discussion in [16], the service model in the underlying queueing system can be approximated by a two-state fluid process, where each state is associated with a "service rate" that corresponds to a channel state. In this context, the service time of a packet refers to the total time needed to *successfully* deliver an information packet to the destination, i.e., it includes all retransmission delays plus the extra time needed to send the FEC parity bits. We assume that the feedback message arrives back at the transmitter before the start of the next transmission slot *of a particular MH*. Let c (in packets/second) be the error-free channel bandwidth that is allocated to the MH, and let c_g and c_b be the service rates that correspond to the Good and Bad states, respectively. Then,

$$c_i = c \cdot e \cdot \left(\frac{1-P_{c,i}}{1-P_{c,i}^{N_l}}\right), \quad \text{for } i \in \{g, b\} \tag{2}$$

In (2), the code rate e accounts for the FEC overhead. This overhead reduces the effective service rate observed at the output of the network buffer. The term $(1-P_{c,i}^{N_l})/(1-P_{c,i})$ is the mean number of retransmissions when the channel is in state i. The appropriateness of the above fluid approximation will be verified in Section VI.

It is easy to see that the evolution of the queue length in the above model is governed by a four-state embedded, continuous-time Markov chain with state space $S = \{(0,g),(0,b),(1,g),(1,b)\}$, where 0 and 1 denote the off and on states of the traffic source, and g and b denote the Good and Bad channel states. In [16] we obtained the queue length distribution for this system and used it to study the packet loss rate due to buffer overflow. In the present work, we start with our previous results in [16] and proceed to derive the distribution for the packet delay. Throughout the paper, matrices and vectors are boldfaced.

Let $\pi_s(x)$ be the CDF of the queue length when the system is in state s at steady-state, $s \in S$, and let $\mathbf{\Pi}(x) \triangleq [\ \pi_{0,g}(x) \quad \pi_{0,b}(x) \quad \pi_{1,g}(x) \quad \pi_{1,b}(x)\ ]$. Following a standard fluid approach (e.g., [2]), we have

$$\mathbf{\Pi}(x) = \sum_{i,\ z_i \le 0} a_i \exp(z_i x)\boldsymbol{\phi_i} \tag{3}$$

where a_i's are constant coefficients and the pairs $(z_i, \boldsymbol{\phi}_i), i = 1, 2, \cdots$, are the eigenvalues and the left eigenvectors of the matrix $\mathbf{MD^{-1}}$; $\mathbf{D} \triangleq \text{diag}[-\mathbf{c_g}, -\mathbf{c_b}, \mathbf{r}-\mathbf{c_g}, \mathbf{r}-\mathbf{c_b}]$ is the drift matrix and $\mathbf{M}$ is the generator matrix of the underlying Markov chain:

$$\mathbf{M} = \begin{bmatrix} -(\beta+\delta) & \delta & \beta & 0 \\ \gamma & -(\beta+\gamma) & 0 & \beta \\ \alpha & 0 & -(\alpha+\delta) & \delta \\ 0 & \alpha & \gamma & -(\alpha+\gamma) \end{bmatrix}.$$

Closed-form expressions for the a_i's, z_i's, and $\phi_{\mathbf{i}}$'s were provided in [16]. The packet loss rate due to buffer overflow is given by

$$G(x) = 1 - \mathbf{\Pi}(x) \cdot \mathbf{1} \tag{4}$$

where $\mathbf{1}$ is a column vector of ones. Let $\boldsymbol{w} \triangleq [w_{0,g} \quad w_{0,b} \quad w_{1,g} \quad w_{1,b}]$ denotes the stationary probability vector of the Markov chain; $\boldsymbol{w}$ satisfies $\boldsymbol{w} \cdot \boldsymbol{M} = \mathbf{0}$ and $\boldsymbol{w} \cdot \mathbf{1} = 1$, resulting in

$$\boldsymbol{w} = \frac{1}{(\alpha+\beta)(\delta+\gamma)} \left[\begin{array}{cccc} \alpha\gamma & \alpha\delta & \beta\gamma & \beta\delta \end{array} \right]. \tag{5}$$

III. Analysis of the Delay Performance

We now derive the delay distribution in the above queueing model. In typical fluid models (for example, the ones used to analyze the performance at an ATM multiplexer [11]), the service rate is constant, making it straightforward to obtain the delay distribution from the queue length distribution. For example, if the channel has only one service rate $\acute{c}$ that corresponds to one BER (i.e., *static* channel), then Pr[delay $\leq t$] is simply given by $\pi_1(\acute{c}t)(\alpha+\beta)/\beta$ (the probability that the queue length as seen by an arrival is $\leq \acute{c}t$), where $\pi_1(.)$ is the queue length distribution when the source is in the on state. However, in our case the channel characteristics are time-varying with two service rates. This makes the delay analysis quite non-trivial since the amount of time that it takes to drain the backlogged traffic seen by an arriving atom of fluid varies depending on the channel state.

Let $c(t)$ be the service rate at time t. We define the steady-state *accumulative service* over a period τ by

$$C(\tau) \triangleq \lim_{t\to\infty} \int_t^{t+\tau} c(u)du$$

Let D be the delay experienced by an atom of fluid at steady-state. The channel state at time t is denoted by $h(t) \in \{g, b\}$, where 'g' and 'b' denote Good and Bad states, respectively. Then,

$$\begin{aligned} \Pr[D \leq \tau] &= \Pr[C(\tau) \geq \widetilde{Q}] \\ &= \sum_{i\in S} \int_{0^-}^{\infty} \Pr[C(\tau) \geq x | i, \widetilde{Q} = x] \left(\frac{d\widetilde{\pi}_i(x)}{dx} \right) dx \quad (6) \\ &= \frac{r}{T} \int_{0^-}^{\infty} \left\{ \Pr[C_g(\tau) \geq x] \left(\frac{d\pi_{1,g}(x)}{dx} \right) \right. \\ &\quad \left. + \Pr[C_b(\tau) \geq x] \left(\frac{d\pi_{1,b}(x)}{dx} \right) \right\} dx \quad (7) \end{aligned}$$

where $\widetilde{Q}$ is the queue length as seen by an arrival at steady-state; $\widetilde{\pi}_i(x) \triangleq \Pr[\widetilde{Q} \leq x,$ system is in state $i]$, $i \in S$; T is the throughput; and $C_i(\tau)$, $i \in \{g, b\}$, is defined by

$$C_i(\tau) \triangleq \lim_{t\to\infty} \left\{ \int_t^{t+\tau} c(u)du \text{ conditioned on } h(t) = i \right\}.$$

In obtaining (7), we made use of the fact that in a fluid system, $\widetilde{\pi}_i(x) = \lambda_i \pi_i(x)/T$, where λ_i is the source arrival rate in state i (in our case, $\lambda_i = 0$ or r) [2]. Hence, $r\pi(x)/T$ represents the fraction of the flow that arrives at the queue when its content is $\leq x$. For an infinite-capacity buffer, the throughput T is given by:

$$T = r(w_{1,g} + w_{1,b}). \tag{8}$$

In order to evaluate (7), we need to determine $\Pr[C_i(\tau) \leq x]$, for $i \in \{g, b\}$. Essentially, the problem is equivalent to determining the distribution for the workload that is generated by a two-state fluid source over a fixed interval of time τ. Let $L_g(\tau)$ and $L_b(\tau)$ be the accumulative sojourn times for the Good and Bad channel states during an interval of length τ:

$$L_i(\tau) \triangleq \lim_{t\to\infty} \int_t^{t+\tau} 1_{\{h(s)=i\}} ds, \quad i = g,\ b$$

where $1_{\{.\}}$ is the indicator function. In Markov theory, the random process $\{L_i(\tau) : \tau > 0\}$ is known as the *occupation time process for state* i [30]. For a given τ, $L_g(\tau)$ ($L_b(\tau)$) represents the portion of the τ interval during which the process is in the Good (Bad) state.

The accumulative service $C(\tau)$ can be written as

$$C(\tau) = c_g L_g(\tau) + c_b L_b(\tau), \quad 0 \leq L_g(\tau), L_b(\tau) \leq \tau. \tag{9}$$

Since $\tau = L_g(\tau) + L_b(\tau)$, $C(\tau)$ can be expressed as

$$C(\tau) = c_g L_g(\tau) + c_b(\tau - L_g(\tau)) = c_g(\tau - L_b(\tau)) + c_b L_b(\tau). \tag{10}$$

Thus, for $c_b\tau \leq x \leq c_g\tau$

$$\Pr[C(\tau) \geq x] = \Pr[L_g(\tau) \geq \frac{x - c_b\tau}{c_g - c_b}] = \Pr[L_b(\tau) \leq \frac{c_g\tau - x}{c_g - c_b}] \tag{11}$$

Note that $c_b\tau \leq C(\tau) \leq c_g\tau$. For a two-state continuous-time Markov chain, the conditional CDFs for the occupation times $L_g(\tau)$ and $L_b(\tau)$ were obtained in [30, page 384]. For $s < \tau$, we have

$$\begin{aligned} \Pr[L_g(\tau) \leq s | h(0) = g] &= \sum_{n=1}^{\infty} e^{-(\delta+\gamma)\tau} \frac{((\delta+\gamma)\tau)^n}{n!} \\ &\cdot \sum_{k=1}^{n} \binom{n}{k-1} \left(\frac{\gamma}{\delta+\gamma}\right)^{k-1} \left(\frac{\delta}{\delta+\gamma}\right)^{n-k+1} \\ &\cdot \sum_{i=k}^{n} \binom{n}{i} \left(\frac{s}{\tau}\right)^{i} \left(1 - \frac{s}{\tau}\right)^{n-i} \\ &= e^{-(\delta+\gamma)\tau} \sum_{n=1}^{\infty} \frac{(\delta\tau)^n}{n!} \sum_{k=1}^{n} \binom{n}{k-1} \left(\frac{\gamma}{\delta}\right)^{k-1} \\ &\cdot \sum_{i=k}^{n} \binom{n}{i} \left(\frac{s}{\tau}\right)^{i} \left(1 - \frac{s}{\tau}\right)^{n-i} \end{aligned}$$

And for $s = \tau$, we have $\Pr[L_g(\tau) = \tau | h(0) = g] = e^{-\delta\tau}$. In a similar way, the probability $\Pr[L_b(\tau) \leq s | h(0) = b]$ for $s < \tau$ is given by:

$$\begin{aligned} \Pr[L_b(\tau) \leq s | h(0) = b] &= e^{-(\delta+\gamma)\tau} \sum_{n=1}^{\infty} \frac{(\gamma\tau)^n}{n!} \\ &\cdot \sum_{k=1}^{n} \binom{n}{k-1} \left(\frac{\delta}{\gamma}\right)^{k-1} \sum_{i=k}^{n} \binom{n}{i} \left(\frac{s}{\tau}\right)^{i} \left(1 - \frac{s}{\tau}\right)^{n-i} \end{aligned}$$

And for $s = \tau$, $\Pr[L_b(\tau) = \tau | h(0) = b] = e^{-\gamma\tau}$. The above expressions along with (11) can be used to obtain $\Pr[C_g(\tau) \geq x]$ and $\Pr[C_b(\tau) \geq x]$:

$$\Pr[C_g(\tau) \geq x] = 1 - \Pr\left[L_g(\tau) \leq \frac{x - c_b\tau}{c_g - c_b} | h(0) = g\right] = 1 - e^{-(\delta+\gamma)\tau} \sum_{n=1}^{\infty} \frac{(\delta\tau)^n}{n!} \sum_{k=1}^{n} \binom{n}{k-1} \left(\frac{\gamma}{\delta}\right)^{k-1} \cdot \sum_{i=k}^{n} \binom{n}{i} \chi^i (1-\chi)^{n-i} \quad (12)$$

$$\Pr[C_b(\tau) \geq x] = \Pr\left[L_b(\tau) \leq \frac{c_g\tau - x}{c_g - c_b} | h(0) = b\right] = e^{-(\delta+\gamma)\tau} \sum_{n=1}^{\infty} \frac{(\gamma\tau)^n}{n!} \sum_{k=1}^{n} \binom{n}{k-1} \left(\frac{\delta}{\gamma}\right)^{k-1} \cdot \sum_{i=k}^{n} \binom{n}{i} \chi^{n-i} (1-\chi)^i. \quad (13)$$

where $\chi = \frac{x - c_b\tau}{(c_g - c_b)\tau}$. The expressions in (12) and (13) involve triple sums, with index in the outer sum ranging from $n = 1$ to ∞. However, we found that this sum converges quite rapidly with almost no change in its value for $n > 30$. Furthermore, the computational time can be significantly reduced by observing the duplicate computations in the last sum for consecutive k's.

Substituting (12), (13), and the derivative of (3) in (7), and after some manipulations, we arrive at the following result, whose proof is given in Appendix A.

Proposition III.1:

$$\begin{aligned}\Pr[D \leq \tau] = {} & \frac{r}{T}(\pi_{1,g}(c_g\tau) + \pi_{1,b}(c_b\tau)) \\ & - \frac{r}{T} e^{-(\delta+\gamma)\tau} \sum_{n=1}^{\infty} \frac{(\delta\tau)^n}{(n+1)!} \sum_{k=1}^{n} \binom{n}{k-1} \left(\frac{\gamma}{\delta}\right)^{k-1} \\ & \cdot \sum_{i=k}^{n} (c_g - c_b)\tau \sum_j a^*_{3j} e^{z_j c_b \tau} \Phi(i+1; n+2; z_j(c_g - c_b)\tau) \\ & + \frac{r}{T} e^{-(\delta+\gamma)\tau} \sum_{n=1}^{\infty} \frac{(\gamma\tau)^n}{(n+1)!} \sum_{k=1}^{n} \binom{n}{k-1} \left(\frac{\delta}{\gamma}\right)^{k-1} \\ & \cdot \sum_{i=k}^{n} (c_g - c_b)\tau \sum_j a^*_{4j} e^{z_j c_b \tau} \\ & \cdot \Phi(n-i+1; n+2; z_j(c_g - c_b)\tau) \end{aligned} \quad (14)$$

where for $i = 1, \ldots, 4$, $a^*_{ij} \triangleq a_j z_j \phi_{ji}$; ϕ_{ji} is the ith element of the jth eigenvector; the sums with index j are taken over $\{j : z_j \leq 0\}$ (the zero and negative eigenvalues); and the function Φ is defined by

$$\Phi(x; y; z) \triangleq \sum_{k=0}^{\infty} \frac{(x)_k}{(y)_k} \frac{z^k}{k!}$$

with $(a)_n \triangleq a(a+1)\cdots(a+n-1)$.

IV. Wireless Effective Bandwidth Subject to a Delay Constraint

From a traffic engineering standpoint, one is often interested in determining the minimum amount of bandwidth that is needed to ensure a desired level of performance. This quantity, which is known as the *effective bandwidth*, is central to service provisioning and admission control in QoS-based packet networks. Extensive research has been conducted on the effective bandwidth in wireline networks (e.g., ATM multiplexers) subject to a packet loss constraint (see [6] and the references therein). Under the same loss constraint, the effective bandwidth has recently been studied for transmission over a wireless link (i.e., variable service rate) [16], [6]. In this paper, we investigate the effective bandwidth over a wireless link *subject to a delay constraint*. More specifically, we define the *wireless effective bandwidth* as:

$$c^* \triangleq \min\{c | c \text{ satisfies } \Pr[\text{delay} > t] = \varepsilon\} \quad (15)$$

where c is the service rate and the pair (t, ϵ) reflects the target delay guarantee. We now obtain c^* in terms of the source, channel, and error control parameters.

In general, a closed-form expression for the effective bandwidth cannot be obtained without approximation. A common approximation is to express the queueing performance in terms of the dominant eigenvalue, z^*, of the underlying fluid system. This approximation becomes exact as the buffer size goes to infinity. In [16] we showed that in the underlying model, the nonzero eigenvalues, including z^*, are given by the roots of a cubic polynomial. Hence, a closed-form expression for z^* is readily available. The same conclusion can be arrived at differently, and perhaps more concisely, by following the development in [6]. To elaborate, let $\{A(t) : t \geq 0\}$ be the accumulative on/off arrival process and let $\{C(t) : t \geq 0\}$ be the accumulative service process. Then, for a stable system, z^* is the unique positive solution of the following equation [6, Page 299]:

$$\Lambda_A(z) + \Lambda_C(-z) = 0 \quad (16)$$

where Λ_A and Λ_C are the Gärtner-Ellis limits for $\{A(t) : t \geq 0\}$ and $\{C(t) : t \geq 0\}$, respectively (defined as $\Lambda_A(z) \triangleq \limsup_{t\to\infty} t^{-1} \log E[\exp(zA(t)]$ and analogously for Λ_C). Since the service process can be treated as a negative-flow arrival process (with input rates $-c_g$ and $-c_b$), both limits are readily available as special cases of the Gärtner-Ellis limit for the two-state fluid source (Equation 9.186 in [6]). Thus,

$$\Lambda_A(z) \triangleq \frac{1}{2}\left(-\alpha - \beta + rz + \sqrt{(\alpha + \beta - rz)^2 + 4\beta rz}\right) \quad (17)$$

$$\Lambda_C(z) \triangleq \frac{1}{2}\Big((\gamma + \delta) + (c_g + c_b)z + \sqrt{(\gamma + \delta - (c_g + c_b)z)^2 - 4(c_g c_b z^2 - \gamma c_g z - \delta c_b z)}\Big) \quad (18)$$

Substituting (17) and (18) in (16), and after some algebraic manipulations, it can be shown that (16) reduces to the same cubic equation in [16], with one of the roots being z^*.

Having obtained z^*, the packet loss and packet delay probabilities over the wireless channel are approximately given by:

$$G(x) \stackrel{\triangle}{=} \Pr[Q > x] = e^{-z^* x} \quad (19)$$

$$\epsilon \stackrel{\triangle}{=} \Pr[\text{delay} > t] = e^{\Lambda_C(-z^*)t} \quad (20)$$

Our interest here is in (20). Given ϵ and t, we proceed as follows. First, we manipulate (20) to arrive at an expression for z^* in terms of the channel parameters (γ, δ, c_g, and c_b) and the delay constraint (t, ϵ). The resulting expression is then substituted for z in (16), noting that $\Lambda_C(-z^*) = \log \epsilon/t$. Then by manipulating (16) we obtain a closed-form expression for the effective bandwidth. Before proceeding, we define the two parameters η_g and η_b:

$$\eta_i \stackrel{\triangle}{=} c_i/c = e\left(\frac{1-P_{c,i}}{1-P_{c,i}^{N_l}}\right), \quad \text{for } i \in \{g,b\} \quad (21)$$

Note that η_g and η_b do not depend on c. Let $\mu \stackrel{\triangle}{=} \log \epsilon/t$. Substituting (18) in (20), we obtain

$$-(\gamma+\delta) - 2\mu - c(\eta_g+\eta_b)z^* = -\Big[(\gamma+\delta+z^*c(\eta_g+\eta_b))^2$$

$$-4(c^2\eta_b\eta_g z^{*2} + \gamma\eta_g c z^* + \delta\eta_b c z^*)\Big]^{1/2} \quad (22)$$

Squaring both sides and rearranging terms, we arrive at the quadratic equation:

$$z^{*2} + Az^* + B = 0 \quad (23)$$

where

$$A = \frac{\gamma}{\eta_b c} + \frac{\delta}{\eta_g c} + \frac{\mu(\eta_g+\eta_b)}{\eta_g\eta_b c}$$

$$B = \frac{\mu^2 + \mu(\gamma+\delta)}{\eta_g\eta_b c^2}$$

In [18] we show that only the larger of the two roots of (23) satisfies (20)[1], i.e.,

$$z^* = \frac{1}{2c}\left\{-\frac{\gamma}{\eta_b} - \frac{\delta}{\eta_g} - \frac{\mu(\eta_g+\eta_b)}{\eta_g\eta_b} + \sqrt{\left(\frac{\gamma}{\eta_b} + \frac{\delta}{\eta_g} + \frac{\mu(\eta_g+\eta_b)}{\eta_g\eta_b}\right)^2 - 4\left(\frac{\mu^2+\mu(\gamma+\delta)}{\eta_g\eta_b}\right)}\right\} \quad (24)$$

Now consider (16) with $\Lambda_C(-z^*)$ set to μ. Substituting (17) in (16), we have

$$\frac{1}{2}\left(-(\alpha+\beta) + rz^* + \sqrt{(\alpha+\beta-rz^*)^2 + 4\beta r z^*}\right) = -\mu \quad (25)$$

[1]Equation (23) can possibly have two positive roots.

Rearranging the above equation with the square root on one side and squaring, we end up with

$$4\mu(\mu-\alpha-\beta) = 4r(\beta-\mu)z^* \quad (26)$$

Replacing z^* in the above equation by its value in (24), we arrive at a closed-form expression for the effective bandwidth subject to a delay constraint:

$$c^* = \frac{r(\beta-\mu)}{2\mu(\mu-\alpha-\beta)}\left\{-\frac{\gamma}{\eta_b} - \frac{\delta}{\eta_g} - \frac{\mu(\eta_g+\eta_b)}{\eta_g\eta_b} + \sqrt{\left(\frac{\gamma}{\eta_b} + \frac{\delta}{\eta_g} + \frac{\mu(\eta_g+\eta_b)}{\eta_g\eta_b}\right)^2 - 4\left(\frac{\mu^2+\mu(\gamma+\delta)}{\eta_g\eta_b}\right)}\right\} \quad (27)$$

Numerical results based on this expression will be provided in Section VI.

V. Packet Discard Rate

As shown in Figure 1, a packet is discarded at the receiver when the number of retransmissions reaches its limit N_l. Such a limit is determined based on the due-date of the transmitted packet. Let $p(n|i)$ denote the probability of n consecutive packet transmission failures given that the channel was in state i at the time of the first transmission attempt, $i \in \{g,b\}$. Observing the recursive structure between consecutive transmissions, we can obtain the following relation:

$$p(n|g) = p(n-1|g)P_{g,g}(\hat{t})P_{c,g} + p(n-1|b)P_{g,b}(\hat{t})P_{c,g} \quad (28)$$

$$p(n|b) = p(n-1|g)P_{b,g}(\hat{t})P_{c,b} + p(n-1|b)P_{b,b}(\hat{t})P_{c,b} \quad (29)$$

where $P_{i,j}(\hat{t}), i,j \in \{g,b\}$, is the probability that the channel state changes from i to j within $\hat{t}$ amount of time. The quantity $\hat{t}$ corresponds to the turnaround time of the packet transmission. The right-hand side of (28) accounts for the event of a transmission failure during a Good channel state followed by $n-1$ consecutive transmission failures, the first of which occurs while the channel is in state i, for $i \in \{g,b\}$. The second equation can be explained in a similar way except that the first transmission failure occurs during a Bad state.

Rearranging the previous equations in a matrix form, we have

$$\begin{bmatrix} p(n|g) \\ p(n|b) \end{bmatrix} = \begin{bmatrix} P_{g,g}(\hat{t})P_{c,g} & P_{g,b}(\hat{t})P_{c,g} \\ P_{b,g}(\hat{t})P_{c,b} & P_{b,b}(\hat{t})P_{c,b} \end{bmatrix} \begin{bmatrix} p(n-1|g) \\ p(n-1|b) \end{bmatrix} \quad (30)$$

with the initial conditions

$$\begin{bmatrix} p(1|g) \\ p(1|b) \end{bmatrix} = \begin{bmatrix} P_{c,g} \\ P_{c,b} \end{bmatrix}.$$

The probabilities $P_{c,g}$ and $P_{c,b}$ were defined in (1).

Solving (30) recursively, we obtain the packet discarding rate (p_d):

$$p_d = [\hat{w}_g \;\; \hat{w}_b] \begin{bmatrix} P_{g,g}(\hat{t})P_{c,g} & P_{g,b}(\hat{t})P_{c,g} \\ P_{b,g}(\hat{t})P_{c,b} & P_{b,b}(\hat{t})P_{c,b} \end{bmatrix}^{N_l-1} \begin{bmatrix} P_{c,g} \\ P_{c,b} \end{bmatrix} \quad (31)$$

where $\hat{w}_g$ and $\hat{w}_b$ are the steady-state probabilities that the channel state is Good and Bad, respectively, *as seen by a packet at the head of the queue in its first transmission attempt.* Note that $\hat{w}_i$ is different from w_i. In particular, $\hat{w}_i$ corresponds to the fraction of the queued traffic that is drained when the channel is in state i. Since the buffer drains continuously as long as the queue is non-empty, $\hat{w}_i$ is equal to the probability that the queue is non-empty times the ratio between the service rate in state i and the throughput, i.e.,

$$\hat{w}_g = \frac{c_g}{T}(w_{0,g} - \pi_{0,g}(0) + w_{1,g} - \pi_{1,g}(0)) \quad (32)$$

$$\hat{w}_b = \frac{c_b}{T}(w_{0,b} - \pi_{0,b}(0) + w_{1,b} - \pi_{1,b}(0)). \quad (33)$$

As for the probabilities $P_{i,j}(t)$, $i,j \in \{g,b\}$, they can be obtained by using Kolmogorov's equation [30]. Thus, we have

$$\begin{aligned} P_{g,g}(t) &= \tfrac{\gamma}{\delta+\gamma} + \tfrac{\delta}{\delta+\gamma}e^{-(\delta+\gamma)t} \\ P_{b,b}(t) &= \tfrac{\delta}{\delta+\gamma} + \tfrac{\gamma}{\delta+\gamma}e^{-(\delta+\gamma)t} \\ P_{g,b}(t) &= 1 - P_{g,g}(t) \\ P_{b,g}(t) &= 1 - P_{b,b}(t). \end{aligned} \quad (34)$$

Substituting the expressions in (33) and (34) into (31), we obtain a closed-form expression for the packet discard rate p_d:

Proposition V.1:

$$\begin{aligned} p_d = {} & \hat{w}_g P_{c,g}(\alpha_0 + \alpha_1(P_{g,g}(\hat{t})P_{c,g} + P_{g,b}(\hat{t})P_{c,b})) \\ & + \hat{w}_b P_{c,b}(\alpha_0(1 + P_{b,g}(\hat{t})P_{c,g}) + \alpha_1 P_{b,b}(\hat{t})P_{c,b}) \end{aligned} \quad (35)$$

where

$$\alpha_0 = \frac{\lambda_1\lambda_2(\lambda_2^{N_l-2} - \lambda_1^{N_l-2})}{\lambda_1 - \lambda_2} \quad (36)$$

$$\alpha_1 = \frac{\lambda_1^{N_l-1} - \lambda_2^{N_l-1}}{\lambda_1 - \lambda_2} \quad (37)$$

and $\lambda_{1,2}$ are the eigenvalues of the square matrix in (31). That is,

$$\lambda_{1,2} = \frac{P_{g,g}(\hat{t})P_{c,g} + P_{b,b}(\hat{t})P_{c,b}}{2} \pm \frac{1}{2}\left[(P_{g,g}(\hat{t})P_{c,g} + P_{b,b}(\hat{t})P_{c,b})^2 + 4(1 - P_{g,g}(\hat{t}) - P_{b,b}(\hat{t}))P_{c,g}P_{c,b}\right]^{1/2}. \quad (38)$$

Proof. By matrix diagonalization, it is easy to obtain the powers of the square matrix in (31).

VI. Numerical Results and Discussion

In this section, we present numerical examples based on our analysis and contrast them against more realistic simulations. The main objectives of our simulations are to: (1) verify the adequacy of the fluid-based approximation of the traffic source and the retransmission mechanism, and (2) check for any significant inaccuracies that may be caused by the approximate computation of the delay performance (recall that the exact expression for the delay distribution involves an infinite sum that needs to be approximated). The main differences between the analysis and the simulations are as follows:

- The ARQ retransmission mechanism is simulated in a realistic manner, whereby a packet is transmitted repeatedly until it is received with no errors or until it reaches the limit on the number of retransmissions. Accordingly, in the simulations the service time of a packet follows a Markov-modulated truncated geometric distribution (as opposed to a Markov-modulated fluid process in the analysis).
- A finite-buffer capacity is used in the simulations, in contrast to the infinite-buffer-capacity assumption in the analysis.
- In the simulations, an integer number of packets is generated during each on period. The size of each packet (before the FEC overhead) is fixed at 53 bytes. This means that the on periods are multiples of the packet transmission time (in the analysis, the duration of the on period can take any nonnegative real value).

For both the analysis and the simulations, we assume that transitions between channel states occur only at the beginning of a packet transmission slot (i.e., channel is slowly varying).

In our experiments, we vary the BER during the Bad state (P_{eb}) and fix the BER during the Good state at $P_{eg} = 10^{-6}$. We set the mean off period to ten times that of the on period. In addition, we take the parameters related to the wireless channel from [14]. We adopt Bose-Chaudhuri-Hocquenghem (BCH) code [22] for FEC . Since we treat the CRC code as part of the payload, the FEC code is applied to 424-bit blocks (i.e., $k = 424$ bits). For each simulation experiment, a sufficient number of independent runs was conducted to ensure tight 95% confidence intervals. To avoid cluttering the figures, we only plot the average values of these runs. Table I summarizes the values of the various parameters used in our experiments. For the parameters c, P_{eb}, v, and N_l, their default values are shown in the parenthesis.

Figures 2 and 3 are meant to validate the adequacy of the analytical approximations by contrasting them with more realistic simulations. Figure 2 shows the complementary delay distribution for three values of c with $P_{eb} = 10^{-2}, \tau = 7$, and $N_l = \infty$. The difference between the analytical and simulation results is negligible in all cases. It should be noted that in QoS-related performance studies, loss and delay probabilities are typically contrasted in the orders of magnitude by which they differ (e.g., 10^{-4} versus 10^{-6}). Differences that are, say, within a half an order of magnitude are considered negligible from the QoS performance standpoint.

The impact of the retransmission limit (N_l) on the delay performance is shown in Figure 3. A highly acceptable agreement is observed between the analytical and simulation results. As N_l increases, $\Pr[D > t]$ also increases at a fixed t. This is expected since increasing N_l extends the total retransmission time of a packet, which in turn increases the packet service time and, consequently, the queueing delay of other packets in the buffer. Note, however, that the

Parameter	Symbol	Value (default)
Source peak rate	r	1 Mbps = 2604.1667 packets/sec
Channel bandwidth	c	$100 - 8000$ packets/sec (1000)
Mean on period	$1/\alpha$	0.02304 sec
Mean off period	$1/\beta$	0.2304 sec
Mean Good channel period	$1/\delta$	0.1 sec
Mean Bad channel period	$1/\gamma$	0.0333 sec
BER in Good channel state	P_{eg}	10^{-6}
BER in Bad channel state	P_{eb}	$10^{-2} - 10^{-5}$ (10^{-2})
Number of correctable bits	v	$0 - 20$ (7)
Maximum number of retransmissions	N_l	$1 - \infty$ (∞)

TABLE I

PARAMETER VALUES USED IN THE SIMULATIONS AND NUMERICAL RESULTS.

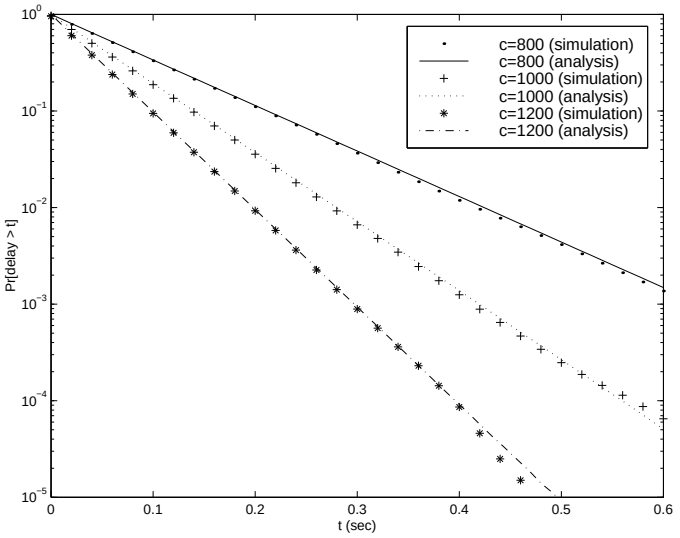


Fig. 2. Complementary delay distribution for different values of c.

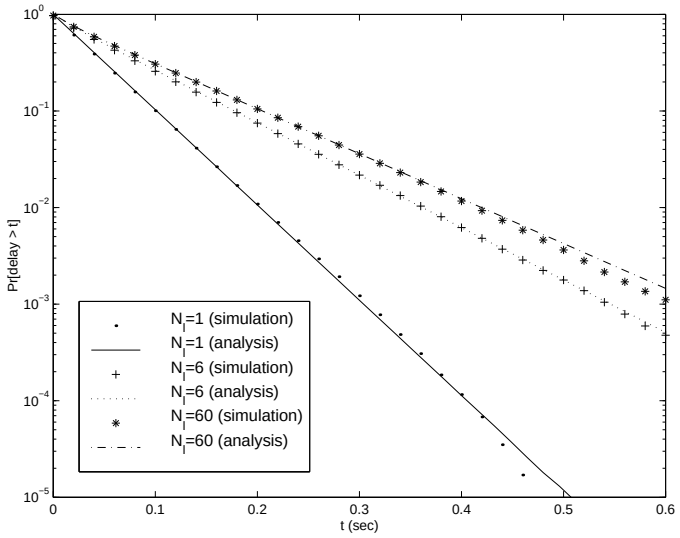


Fig. 3. Complementary delay distribution for different values of N_l.

increase in $\Pr[D > t]$ with N_l is nonlinear; as N_l increases, it starts to have less impact on the delay performance (compare the difference between $N_l = 1$ and $N_l = 6$ with the one between $N_l = 6$ and $N_l = 60$). This trend can be justified based on (2). While increasing N_l increases the queueing delay, is also reduces the PDR at the receiver. Thus, in tuning N_l, one must take into account both the loss and delay requirements of the connection.

Figure 4 depicts the effective bandwidth, computed using (27), versus v (the number of correctable bits per block) for a target delay constraint $\Pr[\text{delay} > 0.1] = \epsilon$, where $\epsilon = 0.0001, 0.01$, and 0.1. A number of observations can be made here. First, the use of hybrid ARQ/FEC with $v > 0$ in more effective from the standpoint of resource allocation than ARQ alone ($v = 0$). In fact, for $\epsilon = 0.0001$, the effective bandwidth at $v = 0$ is almost 45 times its value at $v = 8$ (which happens to be the optimal code rate). The gap tends to decrease as ϵ increases. Second, FEC plays two conflicting roles in the effective bandwidth computation. Increasing the FEC capability reduces the average number of retransmissions per block, leading to higher goodput (and thus, less required bandwidth). But this comes at the expense of increasing the FEC overhead, which in turn reduces the *fraction* of the channel bandwidth that is available for information packets; therefore, increasing the effective bandwidth needed to achieve a given level of QoS. The confluence of the two factors is the reason for the trend observed in the figure, where the effective bandwidth decreases as v increases. This trend continues up to a certain point, v^*, that corresponds to the optimal code rate. Beyond that point, the effective bandwidth starts to increase, indicating that the FEC overhead starts to outweigh its benefits. Third, for $v \geq v^*$ the increase in the effective bandwidth with v is rather slow, which says that it is sufficient to design the FEC encoder to operate within some neighborhood of v^*. This observation can be used in hardware-based adaptive coding schemes, in which the granularity for dynamically adjusting the code rate is limited. One should not, however, undermine the relative increase in the effective bandwidth when the encoder is operating at a v that is significantly higher than v^*. For example, when $\epsilon = 0.01$, going from $v^* = 8$ to $v = 20$ re-

sults in 20% increase in the effective bandwidth. Note that v^* varies from one coding technique to another.

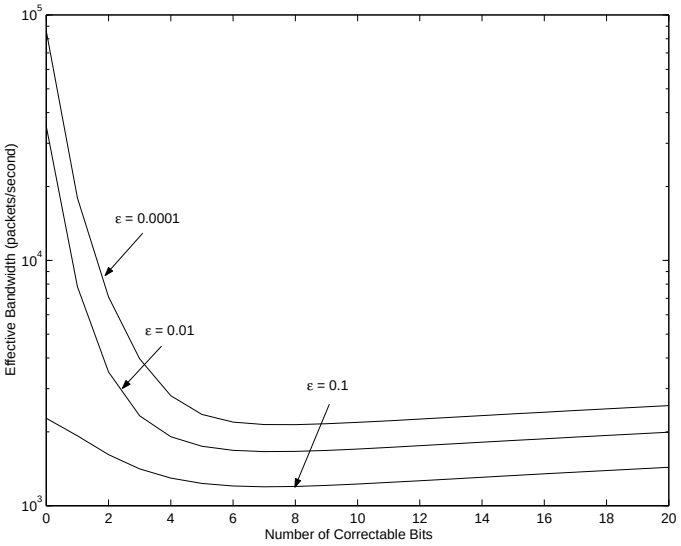


Fig. 4. Effective bandwidth versus v for a target delay requirement $\Pr[\text{delay} > 0.1] = \epsilon$.

Figure 5 depicts the effective bandwidth versus v for four different values of P_{eb} and for a target delay performance of $\Pr[\text{delay} > 0.01] = 0.25$. Varying P_{eb} results in different values for v^*. This implies that different channel environments require different tuning of the FEC code rate *if the encoder is to operate at its optimal code rate.* Interestingly, as v increases, the effective bandwidth becomes less sensitive to P_{eb} (although in this regime, the FEC code rate is non-optimal). So if in a given wireless environment, P_{eb} is hard to estimate accurately or is time-varying, the encoder can be tuned to operate at a sufficiently high FEC code rate (at the expense of extra bandwidth), ensuring a fixed level of delay performance for a constant amount of allocated bandwidth (i.e., independent of the variations in P_{eb}). It is interesting to note that when $P_{eb} = 0.001$, c^* increases almost linearly with v. This can be justified by taking the limit of (27) as P_{eb} approaches P_{eg}, i.e., the channel is always Good. In this case, $c^* = \frac{r(\mu-\beta)}{\eta_g(\mu-\alpha-\beta)}$ For $P_{eg} \ll 1$, $\eta_g \approx e = k/n$. Also, for BCH code with $k = 424$, $n \approx k + 9v$ [22]. Thus,

$$c^* \approx \frac{r(\mu-\beta)}{\mu-\alpha-\beta}(1+\frac{9v}{k}) \qquad (39)$$

which increases linearly with v.

Figure 6 shows the effective bandwidth versus the target delay constraint $\epsilon \stackrel{\triangle}{=} \Pr[\text{delay} > t]$ for $t = 0.001, 0.005, 0.01, 0.1$ seconds and with $v = 7$. At a given t, more bandwidth is expectedly needed to satisfy a more stringent delay requirement, i.e., smaller ϵ. Note that the minimum value of the effective bandwidth (282 packets/second in this example) corresponds to the minimum service rate satisfying the stability condition of the queue, whereas its maximum value (3250 packets/second in this example) is the one satisfying $\min\{c|c_b \geq r\}$, in which case

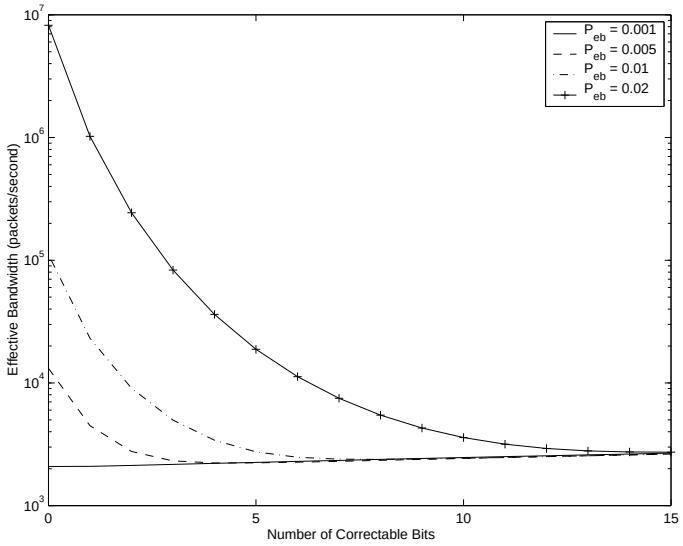


Fig. 5. Effective bandwidth versus v for different values of P_{eb} with $\Pr[\text{delay} > 0.01] = 0.25$.

the queue is always empty. Considering that $r = 2604.1667$ packets/sec, it is worth noting that the *wireless effective bandwidth can indeed be greater than the source peak rate*; a situation that never occurs in wireline networks. The reason behind this phenomenon is that the effective bandwidth in the wireline case depends only on the source characteristics, whereas in the wireless case it also depends on the channel characteristics and the link-layer error control scheme. Another interesting observation is related to the shapes of the plots in Figure 6 at different values of t. When t is small, the effective bandwidth becomes more sensitive to changes in ϵ as ϵ approaches one, while the opposite behavior is observed when t is large.

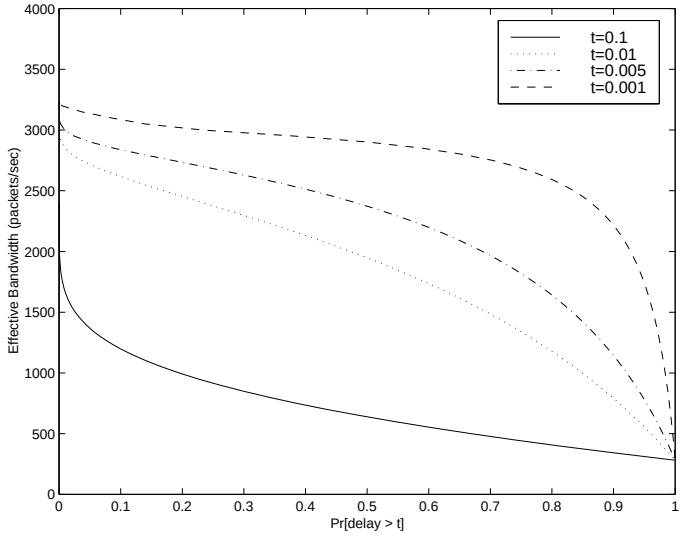


Fig. 6. Effective bandwidth versus $\epsilon = \Pr[\text{delay} > t]$.

In Figure 7 we examine the impact of employing a two-state Markov channel as opposed to a static channel (i.e., independent bit errors with one BER). For the static

channel, the service rate is taken as $\pi_g c_g + \pi_b c_b$, where $[\pi_g \ \ \pi_b] = [\gamma/(\gamma+\delta) \ \ \delta/(\gamma+\delta)]$ are the steady-state probabilities for the Good and Bad states. The figure depicts Pr[delay > 0.5] versus P_{eb} for two values of v (2 and 7) and with the rest of the parameters set to their default values. Several noteworthy remarks can be made here. First, for small values of P_{eb} (e.g., $P_{eb} < 0.005$), there is not much of a difference between the two channel models. However, as P_{eb} increases the delay performance under a 2-state channel model becomes worse than the one obtained under a static channel. This says that *for moderate to high values of P_{eb}, the static channel model underestimates the delay performance* (since it makes sense to think that a 2-state model is closer to a real channel than a 1-state model). The reason behind this behavior is that when $P_{eb} \ll 1$, $1 - P_{c,b} \approx 1 - P_{c,g} \approx 1$, and the two service rates of the 2-state model are almost equal (see (2)). In this case, $c_g(\pi_g + \pi_b) = c_g \approx c_b$. As we increase P_{eb}, the difference between c_g and c_b starts to increase, which consequently increases the variance of the instantaneous service rate and degrades the queueing performance. The point of departure between the two models depends on the value of v. The larger the value of v, the smaller the values of both $P_{c,g}$ and $P_{c,b}$, so a larger P_{eb} would be needed to make $1 - P_{c,b}$ distinctly less than $1 - P_{c,g} \approx 1$. Another interesting observation is related to the behavior of the delay performance for very small and very large values of P_{eb}. In both regimes (and for both channel models), the delay performance seems to be insensitive to the value of P_{eb}. If P_{eb} is very small, it is clear from the above discussion that $c_g \approx c_b \approx ce$, which does not depend on P_{eb}. On the other hand, if P_{eb} is very large (say, $P_{eb} > 0.05$), $P_{c,b} \approx 1$, so that $c_b \approx 0 \ll c_g$, and the performance is mainly determined by c_g, π_g, and π_b. Finally, as we vary P_{eb}, the location of the optimal v shifts accordingly, which explains the difference between the results for $v = 2$ and those for $v = 7$.

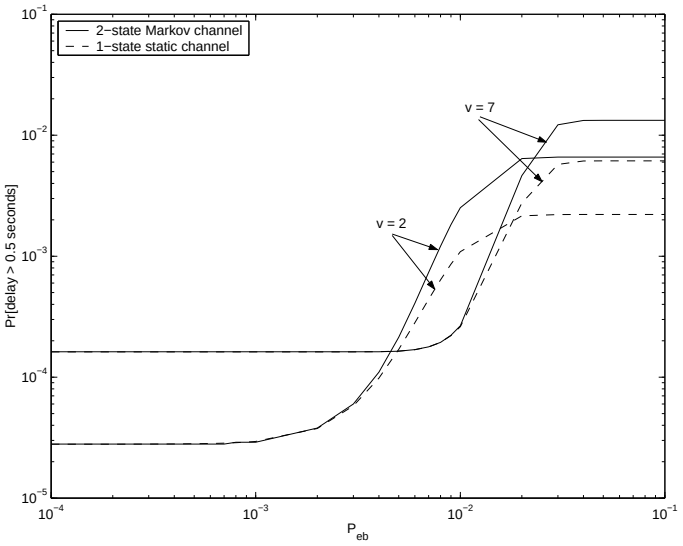


Fig. 7. Pr[delay > 0.5] versus P_{eb} for static and 2-state channel models ($v = 2$ and 7).

VII. Conclusions

In this paper, we presented an approximate, fluid-based analytical model for a wireless link that implements a hybrid ARQ/FEC error control scheme with limited ARQ retransmissions. We used this model to derive the packet delay distribution at the transmitter and the packet discard rate (PDR) at the receiver. In contrast to previous studies, our analysis accommodated both the burstiness of the arriving traffic at the transmitter and the time-varying and autocorrelated characteristics of the radio channel. In deriving the delay distribution, we obtained the distribution for the workload generated by a two-state fluid source (with nonzero generation rates) over a fixed duration of time.

The fluid approximation was a necessary simplification that allowed us to obtain tractable queueing results. The adequacy of this approximation was verified by contrasting the analytical results against more realistic simulations. We then provided a simple closed-form expression for the wireless effective bandwidth subject to a delay constraint. Our results were used to study the interactions among various channel and error control parameters and to assess the impact of these parameters on the QoS performance. We found that in most cases, the use of hybrid ARQ/FEC is more beneficial from resource allocation standpoint than ARQ alone. More importantly, FEC plays dual, and conflicting, roles in the effective bandwidth computation. The confluence of these roles results in the existence of an optimal FEC code rate that produces the minimum effective bandwidth over all possible code rates. We noted that the effective bandwidth decreases rapidly as we increase v from zero (ARQ only) to $v = v^*$ (optimal code rate). Beyond that, the effective bandwidth starts to slowly increase with v, indicating that the overhead of FEC starts to outweigh its benefits. This slow rate of increase can be advantageously used in designing adaptive hardware FEC coders that allow for a limited FEC code-rate settings.

In addition to the code rate, the effective bandwidth also depends on the channel parameters. However, we found that as v increases beyond v^*, the effective bandwidth becomes less sensitive to the value of P_{eb} (with the remaining channel parameters fixed). This means that if one can tolerate a small increase in the allocated bandwidth by operating at a large, non-optimal v, then the QoS performance can be determined independent of P_{eb} (which may be hard to estimate or is time varying). While the effective bandwidth in wireline networks is known to lie between the mean and peak source rates, we found that because of its dependence on the error control and channel parameters, the wireless effective bandwidth may even exceed the source peak rate.

We also examined the impact of employing a two-state channel model as opposed to a (one-state) static model. We observed that depending on the value of P_{eb}, three contrasting behaviors are produced by the two models. In the low range of P_{eb} (e.g., $P_{eb} < 0.001$), both models give comparable results that are insensitive to the value of P_{eb}. However, as P_{eb} increases and moves into its middle range (which

is the one of most practical significance), the two models behave differently, with the static model giving more optimistic results. The point of departure between the two models depends on v; the larger the value of v, the greater the value of P_{eb} at which the two models depart from each other. Also, in this regime, the delay performance is quite sensitive to the value of P_{eb}. As we get into the high range of P_{eb} (e.g., $P_{eb} > 0.05$), the two models give different results but which are insensitive to the value of P_{eb}.

Appendix

I. Proof of Proposition III.1

From (7), we have

$$\begin{aligned}\Pr[D \le \tau] &= \frac{r}{T}\int_{0^-}^{c_b\tau} \{\Pr[C_g(\tau) \ge x]\acute{\pi}_{1,g}(x) \\ &\quad + \Pr[C_b(\tau) \ge x]\acute{\pi}_{1,b}(x)\}\,dx \\ &\quad + \frac{r}{T}\int_{c_b\tau}^{c_g\tau} \{\Pr[C_g(\tau) \ge x]\acute{\pi}_{1,g}(x) \qquad (40)\\ &\quad + \Pr[C_b(\tau) \ge x]\acute{\pi}_{1,b}(x)\}\,dx \qquad (41)\\ &= \frac{r}{T}(\pi_{1,g}(c_b\tau) + \pi_{1,b}(c_b\tau)) \\ &\quad + \frac{r}{T}\left\{\int_{c_b\tau}^{c_g\tau} \Pr[C_g(\tau) \ge x]\acute{\pi}_{1,g}(x)dx \right. \\ &\quad \left. + \int_{c_b\tau}^{c_g\tau} \Pr[C_b(\tau) \ge x]\acute{\pi}_{1,b}(x)dx\right\}. \qquad (42)\end{aligned}$$

where $\acute{\pi}_{1,i}(x) \triangleq \frac{d\pi_{1,i}(x)}{dx}$ for $i \in \{g, b\}$. Consider the first integral in (42). By substituting for $\Pr[C_g(\tau) \ge x]$ from (12), we get:

$$\begin{aligned}\int_{c_b\tau}^{c_g\tau} \Pr[C_g(\tau) \ge x]\acute{\pi}_{1,g}(x)dx &= \pi_{1,g}(c_g\tau) - \pi_{1,g}(c_b\tau) \\ -e^{-(\delta+\gamma)\tau}\sum_{n=1}^{\infty}\frac{1}{n!}\left(\frac{\delta}{c_g - c_b}\right)^n \sum_{k=1}^{n}\binom{n}{k-1}\left(\frac{\gamma}{\delta}\right)^{k-1} \\ \cdot\sum_{i=k}^{n}\binom{n}{i}\int_{c_b\tau}^{c_g\tau}(x - c_b\tau)^i(c_g\tau - x)^{n-i}\acute{\pi}_{1,g}(x)dx. \qquad (43)\end{aligned}$$

Now consider the integral in (43):

$$\begin{aligned}\int_{c_b\tau}^{c_g\tau}(x - c_b\tau)^i(c_g\tau - x)^{n-i}\acute{\pi}_{1,g}(x)dx &= (c_g\tau - c_b\tau)^n \\ \cdot\int_{c_b\tau}^{c_g\tau}\left(\frac{x - c_b\tau}{c_g\tau - c_b\tau}\right)^i\left(1 - \frac{x - c_b\tau}{c_g\tau - c_b\tau}\right)^{n-i}&\acute{\pi}_{1,g}(x)dx \\ = (c_g\tau - c_b\tau)^{n+1}\int_0^1 \chi^i(1-\chi)^{n-i}\acute{\pi}_{1,g}((c_g - c_b)\tau\chi + c_b\tau)d\chi& \qquad (44)\end{aligned}$$

where

$$\chi \triangleq \frac{x - c_b\tau}{\tau(c_g - c_b)}. \qquad (45)$$

From (3), the ith element of the vector $\mathbf{\Pi}(x)$ is given by $\pi_i(x) = \sum_j a_j e^{z_j x}\phi_{ji}$ for $i = 1, \ldots, 4$, where ϕ_{ji} is the ith element of the jth eigenvector and the sum is taken over $\{j : z_j \le 0\}$. Thus, $\acute{\pi}_i(x) = \sum_j a_j z_j \phi_{ji} e^{z_j x} = \sum_j a^*_{ij} e^{z_j x}$, where $a^*_{ij} \triangleq a_j z_j \phi_{ji}$. By our convention, $\pi_{1,g}(x) = \pi_3(x)$. Accordingly, the last integral in (44) reduces to

$$\begin{aligned}\int_0^1 \chi^i(1-\chi)^{n-i}\acute{\pi}_{1,g}((c_g - c_b)\tau\chi + c_b\tau)d\chi = \\ \sum_j a^*_{3j}e^{z_j c_b\tau}\int_0^1 \chi^i(1-\chi)^{n-i}e^{z_j(c_g - c_b)\tau\chi}d\chi. \qquad (46)\end{aligned}$$

By the formula in [31],

$$\int_0^1 x^i(1-x)^{n-i}e^{\beta x}dx = B(n-i+1, i+1)\,\Phi(i+1; n+2; \beta)$$

where $B(x, y)$ is the beta function given by

$$B(x, y) \triangleq \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)} = \int_0^1 t^{x-1}(1-t)^{y-1}dt \qquad (47)$$

and $\Phi(x; y; z)$ is the confluent hypergeometric function given by

$$\begin{aligned}\Phi(x; y; z) &\triangleq \sum_{k=0}^{\infty}\frac{(x)_k}{(y)_k}\frac{z^k}{k!} \\ &= \frac{1}{B(x, y-x)}z^{1-y}\int_0^z e^t t^{x-1}(z-t)^{y-x-1}dt \qquad (48)\end{aligned}$$

with $(a)_n \triangleq a(a+1)\cdots(a+n-1) = \frac{\Gamma(a+n)}{\Gamma(a)}$.

Thus, we have

$$\begin{aligned}\int_0^1 \chi^i(1-\chi)^{n-i}e^{z_j(c_g - c_b)\tau\chi}d\chi = \\ \frac{(n-i)!i!}{(n+1)!}\Phi(i+1; n+2; z_j(c_g - c_b)\tau) \qquad (49)\end{aligned}$$

Now, (43) becomes

$$\begin{aligned}\int_{c_b\tau}^{c_g\tau} \Pr[C_g(\tau) \ge x]\acute{\pi}_{1,g}(x)dx &= (\pi_{1,g}(c_g\tau) - \pi_{1,g}(c_b\tau)) \\ -e^{-(\delta+\gamma)\tau}\sum_{n=1}^{\infty}\frac{(\delta\tau)^n}{n!}\sum_{k=1}^{n}\binom{n}{k-1}\left(\frac{\gamma}{\delta}\right)^{k-1} \\ \cdot\sum_{i=k}^{n}\binom{n}{i}(c_g - c_b)\tau\sum_j a^*_{3j}e^{z_j c_b\tau}\frac{(n-i)!i!}{(n+1)!} \\ \cdot\Phi(i+1; n+2; z_j(c_g - c_b)\tau) \qquad (50)\\ = \pi_{1,g}(c_g\tau) - \pi_{1,g}(c_b\tau) - e^{-(\delta+\gamma)\tau}\sum_{n=1}^{\infty}\frac{(\delta\tau)^n}{(n+1)!} \\ \cdot\sum_{k=1}^{n}\binom{n}{k-1}\left(\frac{\gamma}{\delta}\right)^{k-1}\sum_{i=k}^{n}(c_g - c_b)\tau \\ \cdot\sum_j a^*_{3j}e^{z_j c_b\tau}\Phi(i+1; n+2; z_j(c_g - c_b)\tau). \qquad (51)\end{aligned}$$

In a similar way, the second integral in (42) reduces to

$$\begin{aligned}
&\int_{c_b\tau}^{c_g\tau} \Pr[C_b(\tau) \geq x]\hat{\pi}_{1,b}(x)dx \\
&= e^{-(\delta+\gamma)\tau} \sum_{n=1}^{\infty} \frac{(\gamma\tau)^n}{(n+1)!} \sum_{k=1}^{n} \binom{n}{k-1} \left(\frac{\delta}{\gamma}\right)^{k-1} \\
&= \sum_{i=k}^{n} (c_g - c_b)\tau \sum_j a_{4j}^* e^{z_j c_b \tau} \\
&\quad \cdot \Phi(n-i+1; n+2; z_j(c_g - c_b)\tau) \qquad (52)
\end{aligned}$$

Substituting (51) and (52) into (42), we have

$$\begin{aligned}
\Pr[D \leq \tau] &= \frac{r}{T}(\pi_{1,g}(c_g\tau) + \pi_{1,b}(c_b\tau)) - \frac{r}{T}e^{-(\delta+\gamma)\tau} \\
&\cdot \sum_{n=1}^{\infty} \frac{(\delta\tau)^n}{(n+1)!} \sum_{k=1}^{n} \binom{n}{k-1} \left(\frac{\gamma}{\delta}\right)^{k-1} \\
&\cdot \sum_{i=k}^{n} (c_g - c_b)\tau \sum_j a_{3j}^* e^{z_j c_b \tau} \Phi(i+1; n+2; z_j(c_g - c_b)\tau) \\
&+ \frac{r}{T}e^{-(\delta+\gamma)\tau} \sum_{n=1}^{\infty} \frac{(\gamma\tau)^n}{(n+1)!} \sum_{k=1}^{n} \binom{n}{k-1} \left(\frac{\delta}{\gamma}\right)^{k-1} \\
&\cdot \sum_{i=k}^{n} (c_g - c_b)\tau \sum_j a_{4j}^* e^{z_j c_b \tau} \Phi(n-i+1; n+2; z_j(c_g - c_b)\tau).
\end{aligned}$$

This completes the proof.

Acknowledgments

The authors would like to thank the anonymous reviewers for their constructive comments.

References

[1] M. E. Anagnostou and E. N. Protonotarios. Performance analysis of the selective repeat ARQ protocol. *IEEE Transactions on Communications*, 34(2):127–135, Feb. 1986.

[2] D. Anick, D. Mitra, and M. M. Sondhi. Stochastic theory of a data–handling system with multiple sources. *Bell Syst. Tech. J.*, 61:1871–1894, 1982.

[3] E. Ayanoglu, K. Y. Eng, and M. J. Karol. Wireless ATM: limits, challenges, and protocols. *IEEE Pers. Commun.*, 3(4):18–34, Aug. 1996.

[4] J. B. Cain and D. N. McGregor. A recommended error control architecture for ATM networks with wireless links. *IEEE J. Select. Areas Commun.*, 15(1):16–28, Jan. 1997.

[5] J. Capone and I. Stavrakakis. Achievable QoS and scheduling policies in integrated services wireless networks. *Perform. Eval.*, 27/28(1):347–365, Oct. 1996.

[6] C.-S. Chang. *Performance Guarantees in Communication Networks*. Springer-Verlag, 2000.

[7] C.-S. Chang, K.-C. Chen, M.-Y. You, and J.-F. Chang. Guaranteed quality-of-service wireless access to ATM networks. *IEEE Journal on Selected Areas in Communications*, 15(1):106–118, 1997.

[8] H. Chaskar, T. V. Lakshman, and U. Madhow. TCP over wireless with link level error control: Analysis and design methodology. *IEEE/ACM Transactions on Networking*, 7(5):605–615, Oct. 1999.

[9] S. Choi and K. G. Shin. Predictive and adaptive bandwidth reservation for handoffs in QoS-sensitive cellular networks. In *Proceedings of the ACM SIGCOMM '98 Conference*, pages 155–166, Sept. 1998.

[10] E. O. Elliott. Estimates for error rates for codes on burst-noise channels. *Bell Systems Technical Journal*, 42:1977–1997, Sept. 1963.

[11] A. I. Elwalid and D. Mitra. Statistical multiplexing with loss priorities in rate-based congestion control of high-speed networks. *IEEE Trans. Commun.*, 42(11):2989–3002, Nov. 1994.

[12] R. Fantacci. Queueing analysis of the selective repeat automatic repeat request protocol in wireless packet networks. *IEEE Transactions on Vehicular Technology*, 45(2):258–264, May 1996.

[13] E. N. Gilbert. Capacity of a burst-noise channel. *Bell Systems Technical Journal*, 39:1253–1265, Sept. 1960.

[14] N. Guo and S. D. Morgera. Frequency-hopped ARQ for wireless network data services. *IEEE J. Select. Areas Commun.*, 12(8):1324–1336, Sept. 1994.

[15] I. Joe. An adaptive hybrid ARQ scheme with concatenated FEC codes for wireless ATM. In *Proceedings of MobiCom '97 Conference*, pages 131–138, Sept. 1997.

[16] J. G. Kim and M. Krunz. Bandwidth allocation in wireless networks with guaranteed packet-loss performance. *IEEE/ACM Transactions on Networking*, 8(3):337–349, June 2000.

[17] A. G. Konheim. A queueing analysis of two ARQ protocols. *IEEE Transactions on Communications*, 28(7):1004–1014, July 1980.

[18] M. Krunz and J. G. Kim. Fluid analysis of delay and packet discard performance for QoS support in wireless networks (extended version). Technical report CENG-TR-99-119, University of Arizona, Department of Electrical & Computer Engineering, July 2000. `http://www.ece.arizona.edu/~krunz/papers.html`.

[19] A. M. S. Kumar and D. Klynmyshyn. Characterization of effective bandwidth as a metric of quality of service for wired and wireless ATM networks. In *Proceedings of the IEEE ICC '97 Conf.*, volume 2, pages 1019–1024. IEEE, 1997.

[20] K. Lee. Supporting mobile multimedia in integrated services networks. *ACM/Baltzer Journal of Wireless Networks*, 2(3):205–217, Aug. 1996.

[21] D. A. Levine, I. F. Akyildiz, and M. Naghshineh. A resource estimation and call admission algorithm for wireless multimedia networks using the shadow cluster concept. *IEEE/ACM Transactions on Networking*, 5(1):1–12, Feb. 1997.

[22] S. Lin and D. J. Costello. *Error Control Coding: Fundamentals and Applications*. Prentice Hall, Englewood Cliffs, New Jersey, 1984.

[23] H. Liu and M. El-Zarki. Performance of H.263 video transmission over wireless channels using hybrid ARQ. *IEEE Journal on Selected Areas in Communications*, 15(9):1775–1786, Dec. 1997.

[24] S. Lu, V. Bharghavan, and R. Srikant. Fair scheduling in wireless packet networks. *IEEE/ACM Transactions on Networking*, 7(4):473–489, Aug. 1999.

[25] A. Muir and J. J. Garcia-Luna-Aceves. Supporting real-time multimedia traffic in a wireless LAN. In *Proceedings of the SPIE Multimedia Computing and Networking Conference*, Feb. 1997.

[26] M. Naghshineh and M. Schwartz. Distributed call admission control in mobile/wireless networks. *IEEE Journal on Selected Areas in Communications*, 14(4):711–717, May 1996.

[27] D. Reininger, R. Izmailov, B. Rajagopalan, M. Ott, and D. Raychaudhuri. Soft QoS control in the WATMnet broadband wireless system. *IEEE Pers. Commun.*, 6(1):34–43, Feb. 1999.

[28] D. J. Reininger, R. Izmailov, B. Rajagopalan, M. Ott, and D. Raychaudhuri. Soft QoS control in the WATMnet broadband wireless system. *IEEE Personal Communications Magazine*, 6(1):34–43, 1999.

[29] M. Rice and S. B. Wicker. Adaptive error control for slowly varying channels. *IEEE Transactions on Communications*, 42(2/3/4):917–925, Feb./March/April 1994.

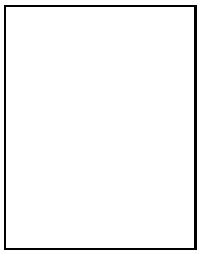
[30] S. Ross. *Stochastic Processes*. John Wiley & Sons, second edition, 1996.

[31] I. M. Ryzhik. *Table of Integrals, Series, and Products*. Academic Press, New York, NY, 1965.

[32] T. Sato, M. Kawabe, T. Kato, and A. Fukasawa. Throughput analysis method for hybrid ARQ schemes over burst error channels. *IEEE Trans. on Vehicular Technology*, 42(1):110–117, Feb. 1993.

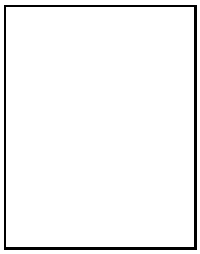
[33] H. S. Wang and N. Moayeri. Finite-state Markov channel — a useful model for radio communications channels. *IEEE Transactions on Vehicular Technology*, 44(1):163–171, Feb. 1995.

[34] M. Zukerman, P. L. Hiew, and M. Gitlits. FEC code rate and bandwidth optimization in WATM networks. In D. Everitt and M. Rumsewicz, editors, *Multiaccess, Mobility, and Teletraffic: Advances in Wireless Networks*, pages 207–220. Kluwer, Boston, 1998.



Marwan M. Krunz received the B.S. in Electrical Engineering from Jordan University, Amman, Jordan, in 1990, and the M.S. and Ph.D. degrees in Electrical Engineering from Michigan State University, Lansing, Michigan, in 1992 and 1995, respectively. From 1995 to 1997, he was a Postdoctoral Research Associate with the Department of Computer Science and the Institute for Advanced Computer Studies at the University of Maryland, College Park. In January 1997, he joined the Department of Electrical and Computer Engineering at the University of Arizona, where he is currently an Assistant Professor. His research interests are in teletraffic modeling, resource allocation in wireless networks, and QoS-based routing.

Dr. Krunz received the National Science Foundation CAREER Award in 1998. He is a Technical Editor for the IEEE Communications Interactive Magazine. He has served and continue to serve on the executive and program committees of several IEEE and ACM conferences.



Jeong Geun Kim received his B.S. and M.S. degrees, both in Electrical Engineering, from Yonsei University, Seoul, Korea, in 1990 and 1992, respectively, and the Ph.D. degree in Electrical and Computer Engineering from the University of Arizona in March 2000. He is currently a Senior Research Staff at Qualcomm Inc., San Jose, CA. His research interests are in wireless networks, quality of service, and packet CDMA systems.