A Simple Packet Transmission Scheme for Wireless Data over Fading Channels *

Heng Wang and Narayan B. Mandayam

Wireless Information Network Laboratory (WINLAB)

Department of Electrical and Computer Engineering

Rutgers University, 73 Brett Road, Piscataway, NJ 08854-8060

Email: hwang, narayan@winlab.rutgers.edu

Abstract

In this paper, we present a simplified scheduling scheme for packet transmission over a fading channel which is modeled as a finite state block channel. We first derive the optimal minimum power transmission policy with constraints on both average delay and packet loss. This problem is seen to be the dual problem of the work by Rajan et. al. [1] where the packet loss rate is minimized under constraints on average delay and power. The optimal policy requires a sophisticated table-look-up for implementation. In order to alleviate this problem, we design a simplified transmission policy that is based on checking for three control parameters: a transmission rate threshold, a channel state threshold and the transmission buffer size. Our results show that the minimum average power with the simplified scheme is very close to that achieved by the optimal policy. By relaxing the packet loss constraint, the simplified policy is also found to allow reduced buffer sizes, thereby simplifying system implementation. With the simplified scheduling policy, the transmitter can be modeled as a bulk service queue and an upper bound for the average delay is derived. Further, the packet loss rate and the average transmit power are estimated using an imbedded Markov chain technique.

*This work is supported in part by the NSF under a CAREER award CCR 9874976 and the New Jersey Commission on Science and Technology under the New Jersey Center for Wireless Communication Technologies. This paper was presented in part at Conference on Information Sciences and Systems (CISS), March 2002, Princeton, NJ.

Keywords: packet scheduling, wireless data, fading channel, dynamic programming, queueing theory

1 Introduction

With the development of personal communication services, portable terminals such as mobile telephones and notebook computers are expected to be used more frequently and for longer times, and hence power consumption will become even more important than it is now. One of the major concerns in supporting such mobile applications is energy conservation and management in mobile devices. Hence, various energy-efficient management techniques have been proposed permeating different protocol layers in wireless data communication systems [2–5].

Current data compression standards have included object-oriented encoding schemes such as MPEG-4 which make wireless multimedia implementable. In these schemes, the source stream consists of several layers of packet streams which may have different Quality of Services (QoS) preferences (such as delay tolerance and packet loss requirement) [6]. Thus, the transmission control strategy has to be determined on the basis of the QoS requirements as well as the dynamics of the packet arriving process. Some earlier work has analyzed the problem of designing a power-efficient transmission schedule for a wireless node in packet data systems. In [7], the effect of traffic burstiness over Gillbert-Elliot channels was studied with a constraint over average delay. Reference [8] explored minimal power transmission of bursty sources for Gaussian channels. Berry and Gallager [10] analyzed the tradeoff between the average delay and the average transmit power in fading environments. In addition, they quantified the behavior of the power-delay tradeoff in the regime of asymptotically large delay. In [1,9], more generalized class of transmission policies are discussed, where some packets are allowed to be dropped besides being transmitted through the channel. The packet is considered lost when the buffer overflows, when it is dropped, or when it is received in error. In [1], Rajan et. al. derived the optimal scheduling policy (packet transmission rate, dropping rate, and transmit power) that minimizes the packet loss with constraints both on the average delay and transmit power. They also discussed a simpler policy where both the packet transmission scheme and the dropping scheme were designed as threshold rules. The parameters of the simplified policy changes for different channel states.

In this paper, we first consider the dual problem of the problem in [1] by minimizing the average transmit power while subject to the constraints on the average delay and the packet loss rate. The major contribution of our work is to propose a much simpler suboptimal policy that performs as well as the optimal policy. The suboptimal policy is determined by only three parameters (a transmission rate threshold, a channel state threshold and the capacity of the transmission buffer) regardless of the number of channel states. The packet loss is controlled solely by provisioning the buffer capacity so that we can neglect the packet dropping scheme in [1]. Further, we also observe that relaxing the packet loss constraint (in a reasonable range) does not help to conserve more power. However, by using the suboptimal policy, relaxing the packet loss constraint can reduce the buffer size notably which substantially simplifies the system implementation.

The common methodology in designing either the optimal or the suboptimal policy is to formulate the average power minimization problem as a constrained Markov decision problem (MDP), which can be solved by dynamic programming (DP) approaches [12]. For a large size buffer, the number of possible system states increase significantly and hence computing the optimal (suboptimal) scheduler is computationally intensive. In addition, adding two constraints (average delay and packet loss) increases the computational complexity further when executing the constrained DP algorithms [13]. In this paper, we model the transmitter with the simplified policy as a single-server bulk service queue and use classical queueing analysis to derive analytical approximations that are precise enough for policy design.

This paper is organized as follows. The finite state block fading channel model and the transmission system model are described in section 2 and 3 respectively. In section 4, the optimal minimum transmission policy is found. Section 5 proposes the simplified sub-optimal policy and section 6 gives the queueing analysis of the transmitter model with the simplified policy.

2 Finite State Block Fading Channel

For high data rate systems, block-interference channel models are generally used to characterize the wireless channel. One block consists of a batch of symbols and these symbols in a block experience the same "channel state". That is to say, the received signal-to-noise ratio (SNR) remains at a constant level for the duration of a block (denoted by $\triangle t$). A finite state block fading channel model can be built as follows. In each block, the channel is modeled as an AWGN channel, i.e.,

$$y = \sqrt{h}x + n \tag{1}$$

where x and y are input and output signals respectively. n is noise, and h is the fading factor. Normalizing the path loss to 1, for the Rayleigh fading channel, h is distributed exponentially with probability density function

$$f_h(x) = e^{-x}, \quad x \geq 0, \tag{2}$$

Let $0 = h^{(0)} < h^{(1)} < \cdots < h^{(K)} = \infty$ be a sequence of pre-selected thresholds, by which we partition fading factor h into a finite number of intervals. Then the channel is said to be in state $s^{(k)}$ if $h \in [h^{(k)}, h^{(k+1)})$, $k = 0, 1, \cdots, K-1$. Let $\mathcal{H}$ denote the channel state set, i.e., $\mathcal{H} = \{s^{(0)}, s^{(1)}, \cdots, s^{(K-1)}\}$. The steady state probability is given as

$$q_{s^{(k)}} = \int_{h^{(k)}}^{h^{(k+1)}} f_h(x)\, dx, \quad s^{(k)} : h \in [h^{(k)}, h^{(k+1)}) \tag{3}$$

A *memoryless*[1] channel model is assumed in this work, where $s^{(k)}$ is i.i.d. in consecutive blocks. Further, we assume both the transmitter and the receiver have the perfect channel state information.

If the power of background noise is also normalized to 1, and if we let R represent the number of data packets transmitted in one block on state $s^{(k)}$, then the minimum power required for error-free reception is given as

$$P_m(s^{(k)}, R) = \frac{1}{h^{(k)}} \left(2^{\frac{2S_p}{S_b} R} - 1 \right), \quad s^{(k)} : h \in [h^{(k)}, h^{(k+1)}) \tag{4}$$

where S_b is the number of information bits being transmitted in one block and S_p is the packet size in bits. Note that the expression in (4) assumes that the number of bits S_b is a fairly large such that the maximum mutual information in one block can be approximated by the channel capacity. Without loss of generality, we also assume the fraction $\frac{S_p}{S_b}$ happens to be an integer. Since only finite channel state information is fedback, we only know h belongs to a interval $[h^{(k)}, h^{(k+1)})$ instead of the exact value. Therefore, the lower threshold $h^{(k)}$ of interval $[h^{(k)}, h^{(k+1)})$ is used to calculate the transmit power because with this power, R packets can be correctly received for sure by the optimal encoding.

3 System Model

Let time t be quantized by block duration $\triangle t$. The system model is shown in Figure 1. Assume that a buffer with a finite capacity of L packets is used to store the incoming packets in the transmitter. An information source sends packets into the buffer with rate λ (packets per block). The number of

[1] In fact, the assumption of a memoryless channel is necessary only when using queueing theory to analyze the scheduling behavior as will be shown later.

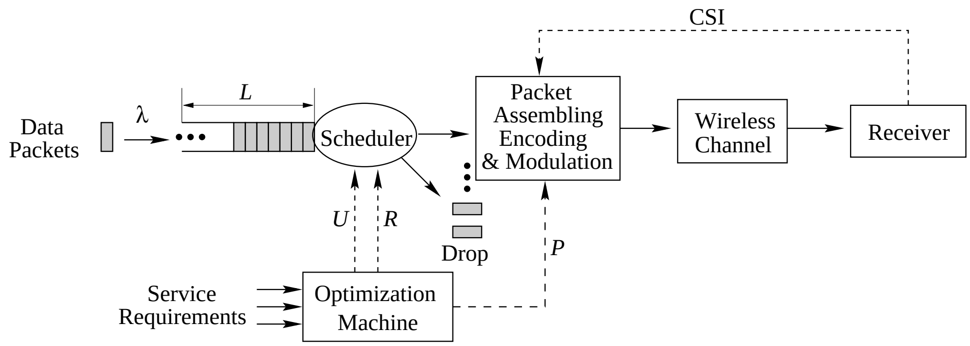


Figure 1: System Model

the arriving packets in block $[t_i, t_{i+1})$ denoted by θ_i, is assumed to be i.i.d. At each time instant t_i ($i = 0, 1, \cdots$), the transmitter withdraws U_i packets from the buffer. In the coming block $[t_i, t_{i+1})$, R_i packets ($R_i \leq U_i$) are assembled, encoded, and transmitted with power P_i through a wireless channel, where U_i, R_i and P_i are determined by a scheduling policy. If the packets transmitted in this block can not be correctly decoded by the receiver, we assume they are lost and no further retransmission will be scheduled. In other words, an *outage* occurs. Note that the packets can be correctly received if the inequality

$$P_i \geq P_m(s_i, R_i) \tag{5}$$

can be met, where s_i is the channel state in the i-th block, and $P_m(s_i, R_i)$ is given in (4). Let the queue length x_i denote the number of packets in the buffer at time instant t_i and set $\mathcal{X} = \{0, 1, \cdots, L\}$. The buffer dynamics is given by the following:

$$x_{i+1} = \min\{\, x_i - R_i + \theta_i,\ L\,\} \tag{6}$$

When the buffer is full, i.e., $x_i = L$, the incoming packets will get *blocked* and can not be recovered. Here we denote *packet loss* to include the following situations: the packet either being dropped at the transmitter, or being corrupted in the air channel, or being blocked when the buffer overflows.

4 Optimal Power and Rate Control

The objective in this section is to find the optimal scheduling policy that minimizes the average transmit power under an average delay constraint and a packet loss constraint. Thus, at each time instant t_i, we need to decide the withdrawal rate U_i, transmission rate R_i and the transmit power P_i for block $[t_i, t_{i+1})$. Considering the delay and the packet loss constraints, U_i, R_i and P_i need to be determined based on both the channel state s_i as well as the queue length x_i. We define

Definition 1 *The system state space* $\mathcal{S} \stackrel{\triangle}{=} \{\mathbf{v} = (x, s)\}$, *where* $x \in \mathcal{X}$ *and* $s \in \mathcal{H}$.

Assume the system is on state $\mathbf{v}_i$ at time t_i. We now define the following three functions

$$U_i = u_i(\mathbf{v}_i), \quad R_i = r_i(\mathbf{v}_i) \quad \text{and} \quad P_i = p_i(\mathbf{v}_i). \tag{7}$$

where $u_i, r_i : \mathcal{X} \times \mathcal{H} \mapsto \{0, \mathbf{Z}^+\}$ and $p_i : \mathcal{X} \times \mathcal{H} \mapsto \{0, \mathbf{R}^+\}$, where $\mathbf{Z}^+$ and $\mathbf{R}^+$ represent positive integer and real numbers respectively. A *control policy* π is defined as a sequence of the triple vector $[u_i, r_i, p_i]$, i.e, $\pi = \{[u_0, r_0, p_0], [u_1, r_1, p_1], \cdots, [u_i, r_i, p_i], \cdots\}$. For an average cost problem with a finite state and control space it is known that there always exists a stationary policy which is optimal [12]. Thus, only stationary policies are considered in this work, which means the control policy does not depend on system time. Therefore equation (7) can be rewritten as

$$U_i = u(\mathbf{v}_i), \quad R_i = r(\mathbf{v}_i) \quad \text{and} \quad P_i = p(\mathbf{v}_i). \tag{8}$$

For a given policy π, assume $\omega(\mathbf{v})$ denotes the steady state probability of state $\mathbf{v}$. Then, the average power is given by

$$\bar{P} = \sum_{\mathcal{S}} \omega(\mathbf{v}) p(\mathbf{v}). \tag{9}$$

and the average dropping rate is

$$P_D = \frac{1}{\lambda} \sum_{\mathcal{S}} \omega(\mathbf{v})[u(\mathbf{v}) - r(\mathbf{v})] \tag{10}$$

The packet block rate (when the buffer overflows) is given as

$$P_B = \sum_{\mathcal{H}} \omega(\mathbf{v} = (L, s)) \tag{11}$$

Thus, the real incoming packet rate accepted by the buffer is $\lambda(1 - P_B)$. By Little's Theorem, the average delay is given as

$$\bar{D} = \frac{1}{\lambda(1 - P_B)} \sum_{\mathcal{S}} x\omega(\mathbf{v}) \tag{12}$$

The average outage probability is given as

$$P_{out} = \frac{1}{\lambda} \sum_{\mathcal{S}} \omega(\mathbf{v})\, \mathbf{1}\Big(p(\mathbf{v}) < P_m(s, r(\mathbf{v}))\Big) \tag{13}$$

where $\mathbf{1}(\cdot)$ is the indicator function, that is $\mathbf{1}(x) = 1$ if x is true and 0 if x is false. Finally, the packet loss rate is defined as $P_{out} + P_D + P_B$. The optimum minimum power problem can now be formally stated as follows.

Problem A

$$\min_{\pi} \bar{P} \tag{A.1}$$

$$\text{subject to} \quad \bar{D} \leq D_{avg} \tag{A.2}$$

$$\text{and} \quad P_{out} + P_D + P_B \leq \eta \tag{A.3}$$

Note that Problem A is the dual problem of the optimization problem in [1].

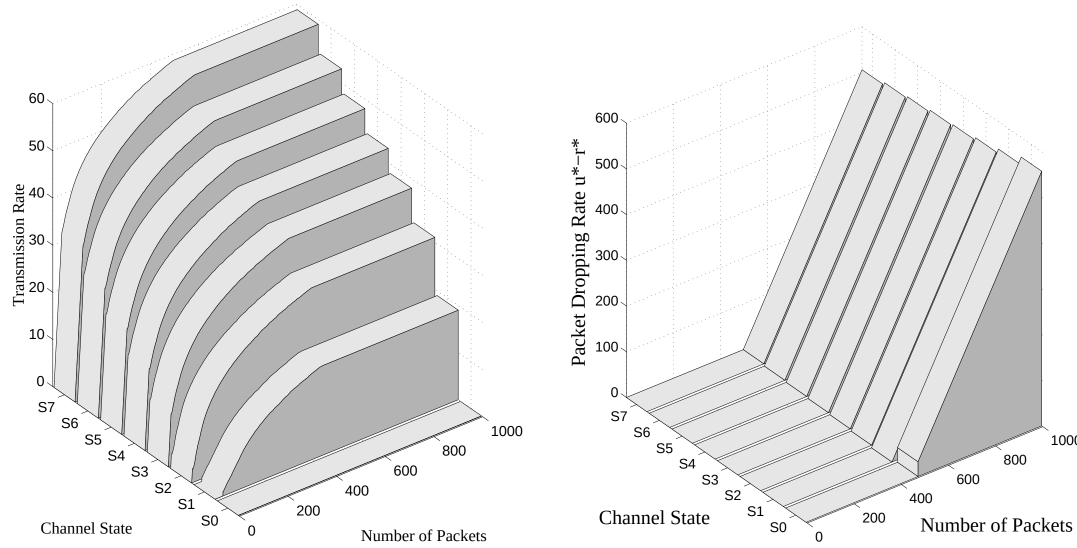
Theorem 1 *Assume policy $\pi^* = [u^*, r^*, p^*]$ is one solution of Problem A, it follows*

$$p^*(\mathbf{v}) = P_m(s^{(k)}, r(\mathbf{v})) \tag{14}$$

Since the proof is very similar with the proof of Proposition 1 in [1], the proof is not given here. The above theorem shows that the transmit power is always chosen to be the minimum required power, i.e., $P_{out}(\pi^*) = 0$. In other words, in order to conserve the power, to drop packets directly is more efficient than to transmit them (but without enough power) in an outage, because those packets will get lost anyway. Therefore, we need to now determine only two functions $u(\mathbf{v})$ and $r(\mathbf{v})$ that solve Problem A. A DP algorithm can be used to solve Problem A. We need to notice that in the discrete system, the recursive relationship of the queue length in equation (6) assumes that the packets arrive right at the time instant t_i, $i = 0, 1, \cdots$. However, packets arrive in a continuous manner during the block interval. If we assume the packet arriving process is independent of channel variation, the real average packet buffered delay is half block less than the average delay we obtain by the DP algorithm. Due to this reason, in all the numerical examples, this delay has to be compensated for.

5 A Simplified Suboptimal Control Scheme

There are some drawbacks of the optimal policy of Problem A:



(a) $r^*(\mathbf{v})$ (b) $u^*(\mathbf{v}) - r^*(\mathbf{v})$

Figure 2: Optimal Policies with D_{avg}=1.5, $\eta = 10^{-3}$

1. The DP algorithm is tedious when the buffers size is big;
2. It requires a sophisticated table look-up for implementation;
3. The dropped packets will not be received at the receiver, but they also contribute to the average queue length of the transmitter buffer. Therefore, the average buffered delay calculated by Little's Theorem cannot precisely characterize the end-to-end delay at the receiver.

We now present the following preliminaries that will lead us to a simplified policy.

Theorem 2 *If u^* and r^* are one solution of Problem A, then*

1. *For a memoryless channel, r^* is non-decreasing in channel state s.*
2. *u^* is non-decreasing in queue length x.*
3. *u^* and r^* have the following relationship*

$$r^*(\mathbf{v}) = \begin{cases} u^*(\mathbf{v}), & r^*(\mathbf{v}) < R_u(s,\eta); \\ R_u(s,\eta), & r^*(\mathbf{v}) \geq R_u(s,\eta). \end{cases} \quad (15)$$

where $R_u(\cdot,\cdot)$ is a value depending on the channel state s and the packet loss constraint η.

The proof of Theorem 2 is given in the Appendix. According to Theorem 2 and based on numerical examples (shown in Figure 2), we make the following statements:

1. $r^* = 0$ for some low SNR states;
2. For high SNR states, $r^* = x$ (the queue length) when x is small. In other words, when there are only few packets in the buffer, the optimal policy is to transmit all of them when the channel quality is good.
3. For each channel state s, there exists a threshold value of queue length x_{th} where $r(x_{th}, s) = R_u(s, \eta)$. Thus when there are less than x_{th} packets in the buffer, no packet is dropped. However, when there are more than x_{th} packets, some packets have to be dropped but the optimal transmission rate stays at a constant $R_u(s, \eta)$ which is not related to the queue length.
4. When $x > x_{th}$, the optimal dropping rate $u^* - r^*$ is almost a linear function (with slope 1) of queue length x. That is to say, $u^* - r^* \approx x - x_{th}$ for $x > x_{th}$.

Based on the above observations, we select three parameters: a fading channel state threshold h_a which is one of thresholds $h^{(k)}$, $k = 0, 1, \cdots, K-1$; a transmission rate threshold r_a, and a queue length threshold for dropping packets x_L. Both r_a and x_L are chosen to be integers. Consider the following transmission rate function:

$$r_s(\mathbf{v} = (x, s^{(k)})) = \begin{cases} 0, & h^{(k+1)} < h_a; \\ x, & h^{(k+1)} \geq h_a \text{ and } x \leq r_a; \\ r_a, & h^{(k+1)} \geq h_a \text{ and } x \geq r_a. \end{cases} \tag{16}$$

According to (16), the transmitter only transmits when the channel is good enough (i.e., $h \geq h_a$). When there are a small number of packets in the buffer ($x < r_a$), the transmitter transmits all of them. When the number of the buffered packets is more than r_a, the transmitter transmits r_a packets in each block. Further a packet withdrawal scheme is defined as

$$u_s(\mathbf{v}) = \begin{cases} r_s(\mathbf{v}), & x < x_L; \\ x + r_a - x_L, & x \geq x_L \end{cases} \tag{17}$$

where we choose $x_L \geq r_a$. Figure 3 shows the conceptual curves of u_s and r_s comparing them with the optimal policy on a particular channel state. According to u_s, when the number of packets is less than x_L, no packet will be dropped. Assuming at time t_i there are more than x_L packets, i.e., $x > x_L$,

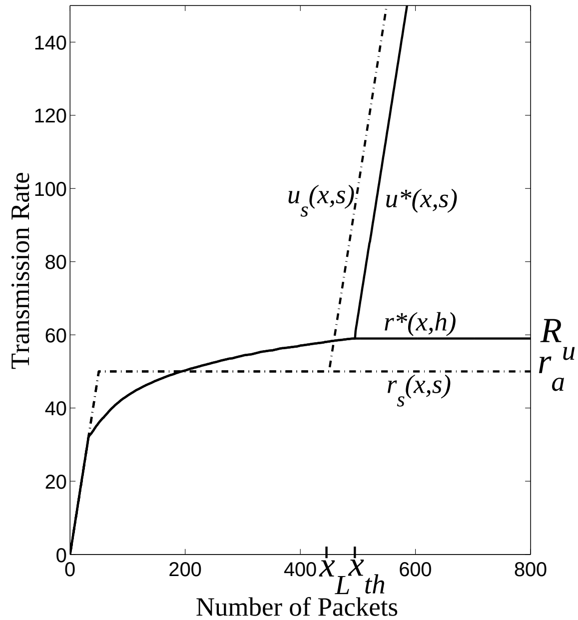


Figure 3: A Simplified Policy

the transmitter will withdraw $u_i = x_i + r_a - x_L$ packets. Let x_{i^+} denote the number of packets at time t_i^+, i.e., right after applying the policy, then

$$x_{i^+} = x_i - u_i = x_L - r_a \tag{18}$$

Equation (18) implies that by applying policy u_s, $x - (x_L - r_a)$ packets are withdrawn and the queue length is kept below $x_L - r_a$. Assuming there is no priority difference between packets, there is no difference on the QoS metrics between dropping packets at the packet arriving end of the buffer, or dropping them at the server (buffer withdrawal) end. We can now define a policy π_s as follows.

Definition 2 *Assume the capacity of the buffer is L packets. On each state $\mathbf{v} = (x, s)$, define a policy π_s such that the transmission rate function $r_s(\mathbf{v})$ is as given in equation (16). Further let the minimum required power for reliable transmission given in (4) be chosen to be the transmit power.*

According to π_s, the packet will get lost only when the buffer overflows, which means the packet block rate is equivalent to the packet loss rate. Note that π_s is equivalent to the policy $[u_s, r_s]$ if L is chosen as $x_L - r_a$. Comparing the suboptimal policy proposed in [1], the advantages of π_s are listed below.

1. π_s depends on only 3 parameters: h_a, r_a and L regardless of the number of the channel states, which is much easier to be implemented.

2. Since no packet is allowed to be dropped, the buffered delay calculated through Little's Theorem is precise.

3. The packet loss rate is controlled only via the buffer capacity L. Hence, for a relatively relaxed packet loss constraint, we do not have to use a huge buffer to avoid buffer overflow. It will greatly simplify the complexity of the DP algorithm as well as the system implementation.

4. π_s is simple enough to be analyzed using classical queueing theory. We will find a relatively easy method to estimate the QoS metrics in the next section which simplifies the system design in comparison to using the DP approach.

We can now formally state the minimum average power problem with the simplified policy as follows.

Problem B

$$\min_{h_a, r_a, L} \bar{P} \tag{B.1}$$

$$\text{subject to} \quad \bar{D} \leq D_{avg} \tag{B.2}$$

$$\text{and} \quad P_B \leq \eta \tag{B.3}$$

The solution of Problem B is the simplified scheme that results in a suboptimal solution to Problem A.

To illustrate the performance of the suboptimal policy, we assume the number of arriving packets in one block obeys a Poisson distribution, i.e., the probability of j packets arriving in one block is given as

$$v_j = \Pr\{\theta = j\} = e^{-\lambda} \frac{\lambda}{j!}, \quad j = 0, 1, 2, \cdots . \tag{19}$$

Consider a packet stream with rate 500 Kbps. If the block duration is assumed to be 1 ms and packet size S_p=100 bits/packet, then $\lambda = 5$ packets/block. Further, we choose S_b=2000 bits/block. An 8-state ($K = 8$) block fading channel model is used by partitioning the channel into SNR intervals with identical steady state probabilities as follows: $S^{(0)}$=($-\infty$, -8.47dB), $S^{(1)}$=[-8.47dB, -5.41dB), $S^{(2)}$=(-5.41dB, -3.28dB), $S^{(3)}$=(-3.28dB, -1.59dB), $S^{(4)}$=(-1.59dB, -0.08dB), $S^{(5)}$=(-0.08dB, 1.42dB), $S^{(6)}$=(1.42dB, 3.18dB), $S^{(7)}$=(3.18dB, ∞).

Figure 4 shows the average transmit power under both the suboptimal policy and the optimal policy (which serves as a lower bound) varying with the average delay constraint when the packet loss is constrained under 10^{-7} and 10^{-3} respectively. It is observed that when the average delay constraint is small, the transmit power under the suboptimal policy is very close to the power under the optimal

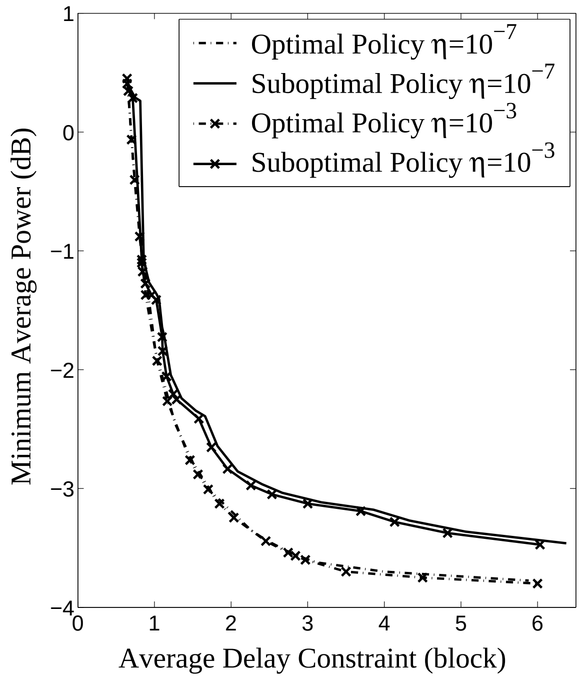


Figure 4: Minimum Power vs. Average Delay

policy. When the delay constraint is large, the difference of the consumed power is less than 0.5dB. This suggests that the suboptimal scheduling policy works effectively.

If we write the average transmit power $\bar{P}(\pi^*|\eta)$ as an explicit function of the packet loss constraint η, it is easy to show that with the same delay constraint,

$$\bar{P}(\pi^*|\eta_1) \geq \bar{P}(\pi^*|\eta_2), \quad \text{if } \eta_1 \leq \eta_2 \tag{20}$$

because with bigger η, more packets can be dropped in order to save the power. However, by observing the numerical examples in Figure 4, we find $\bar{P}(\pi^*|10^{-7}) \approx \bar{P}(\pi^*|10^{-3})$. Similar results are also observed when $\eta \leq 10^{-2}$. It implies that allowing packet loss is not an effective technique to save the transmit power. However, in the above numerical examples, when $D_{avg} < 10$ (blocks), with $\eta = 10^{-7}$, the minimum required buffer size $L^* \approx 1000$, while $L^* < 100$ if $\eta = 10^{-3}$. Based on the above observation, we can conclude that by using the simplified scheduling policy, the required buffer capacity decreases significantly as the packet loss increases.

6 Queueing Analysis

In this section, we use classical queueing results to analyze the scheduling behavior of the simplified policy π_s. With the assumption of a Poisson distribution on the packet arriving process, the transmission system can be modeled as a slotted $M|D^B|1|L$ queueing model [14], which is a single server system with Poisson arriving customers, bulk serving ability (with maximum batch size of r_a packets) for constant service time ($\triangle t$), and with finite buffer size (L). In this slotted $M|D^B|1|L$ model, the slot corresponds to the channel block, i.e., the server (transmitter) withdraws the packets only at the specified time instant t_i, $i = 0, 1, \cdots$. At the moment when the server can start with the service, if the queue length is more than r_a packets, r_a packets will be served at the same time; but if the queue length is less than r_a, the server will start serving all the buffered packets immediately instead of waiting till r_a packets are available. The packets that arrive during the service time have to wait for the next serving slot. The distribution of service time of every batch of packets is deterministic, i.e, one block time. Further, since the transmitter does not transmit any packet when the fading status $h < h_a$, we regard the server to be in a mandatory vacation under this condition. Note that the mandatory vacation that arises here is different from the classical notion of vacation in queueing analysis [16]. Let M denote the duration of a mandatory vacation and it is an integer multiple of the block time $\triangle t$. Since the channel is assumed memoryless, M obeys a geometric distribution as

$$\Pr\{M = m\triangle t\} = \alpha(1-\alpha)^m, \quad m = 0, 1, \cdots \tag{21}$$

where $\alpha = \Pr[h > h_a] = e^{-h_a}$ for a Rayleigh fading channel. The average vacation time $\bar{M} = \frac{1-\alpha}{\alpha}$ blocks. Let F denote the number of packets that arrive during a vacation interval, then

$$\Pr\{F = j\} \triangleq f_j = \sum_{m=0}^{\infty} \left\{ \alpha(1-\alpha)^m e^{-\lambda m} \frac{(\lambda m)^j}{j!} \right\}, \quad j = 0, 1, \cdots \tag{22}$$

Note that the probability of no packet arriving during the vacation is

$$f_0 = \alpha \sum_{m=0}^{\infty} (1-\alpha)^m e^{\lambda m} \tag{23}$$

Figure 5 shows a sample realization of the three different functional system states: “busy” (i.e.,transmitting packets), “idle” (i.e., empty buffer) and “vacation” (i.e., waiting).

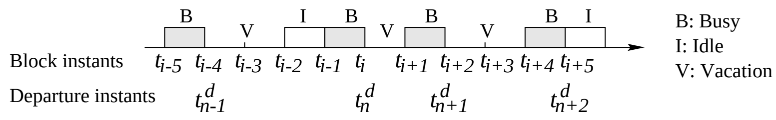


Figure 5: The Functional States of the Transmitter

6.1 An Upper Bound for the Average Delay

We search for an upper-bound for the average delay using the concept of *mean residual service time* [15]. The residual service time here refers to the time span from the arriving instant of a specific marked packet to the beginning instant of the next busy block. Let W denote this residual service time seen by the packet in consideration and D represent the buffered delay of this packet. Note that both W and D here are random variables. The delay D can be written as follows:

$$D = W + V + \left\lfloor \frac{x}{r_a} \right\rfloor \tag{24}$$

where V refers to the number of vacation blocks which the server experiences before the marked packet gets served. x is the queue length seen by this packet upon arrival. $\lfloor \frac{x}{r_a} \rfloor$ represents the nearest integer which is smaller than $\frac{x}{r_a}$, which denotes the number of busy blocks before serving the marked packet. Since function $\lfloor \cdot \rfloor$ is nonlinear, it is very difficult to derive the average delay $\bar{D}$ directly. Hence, we approximate the system as the following simplified system: when the system is in the busy state (i.e., busy periods as shown in Figure 5), we model the system as a $M|D|1|L$ queue where the packets are served one by one with a constant service time $\frac{1}{r_a}$ (in units of blocks). However, when the system is in the idle state or in a vacation, we still treat the system as a slotted system where a slot corresponds to a block. Since $\left\lfloor \frac{x}{r_a} \right\rfloor \leq \frac{x}{r_a}$, it follows that

$$D \leq W + V + \frac{x}{r_a}. \tag{25}$$

By taking expectations on both sides, we obtain

$$\bar{D} \leq \bar{W} + (1-\alpha)(\bar{D} - \bar{W}) + \frac{\bar{x}}{r_a} \tag{26}$$

where $\bar{W}$ is the mean residual time and $\bar{x}$ is the average queue length observed by arriving packets. Note that the average vacation time during the waiting period is given as $\bar{V} = (1-\alpha)(\bar{D} - \bar{W})$. With the assumption of Poisson arrivals, according to Little's Theorem,

$$\bar{x} = \lambda(1 - P_B)\bar{D} \leq \lambda\bar{D} \tag{27}$$

From (26), the upper bound for the average waiting time is given as

$$\bar{D} \leq \frac{\bar{W}}{1-\rho} \tag{28}$$

where $\rho = \frac{\lambda}{\alpha r_a}$. The mean residual time $\bar{W}$ can be approximated by

$$\bar{W} \approx \frac{\lambda}{2r_a^2} + (\frac{1}{2} + \frac{1-\alpha}{\alpha})(1 - \alpha\rho) \tag{29}$$

which is derived in Appendix B. Note that when $r_a = 1$ and $L = \infty$, the equality in (28) holds. If we choose to transmit only one packet in each block (i.e., $r_a = 1$) and transmit on all channel states (i.e., $h_a = 0$ which results in $\alpha = 1$), the average delay reduces to the well known P-K formula [15].

Since the average delay decreases as r_a increases, it implies that the average delay approaches its minimum as $r_a \to \infty$. This corresponds to the situation where in every busy block, the transmitter transmits all the packets in the buffer. Therefore, the average delay is actually the average waiting time to the next busy block, i.e.,

$$\bar{D}_{min} = \lim_{r_a \to \infty} \bar{D} = \frac{1}{2} + \frac{1-\alpha}{\alpha} \tag{30}$$

Figure 6 shows the upper bound of the average delay varying with the transmission rate threshold r_a with a fixed channel state threshold h_a. We also plot the average delay obtained by running a DP algorithm and by a system simulation (a Monte Carlo simulation of the system shown in Figure 1 with the simplified policy). It is observed that the average delay obtained via the DP algorithm is accurate in comparison with that obtained through the simulations. Moreover, the upper bound is less than 0.5 block time higher than the actual delay. It suggests that given the control parameters of the simplified policy (r_a, h_a and L), the average delay can be directly approximated using equation (28).

6.2 Packet Loss Rate and Average Transmit Power

The packet block rate P_B (i.e., the packet loss rate for the simplified policy) and the average transmit power $\bar{P}$ can be obtained via equations (9) and (11). The steady state probability $\omega(\mathbf{v})$ in (9) and (11) can be calculated through dynamic programming which has been used to generate all the numerical

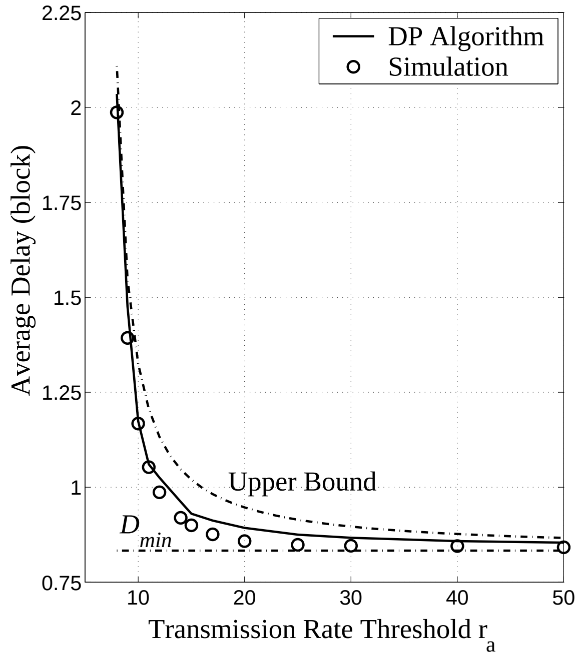


Figure 6: Average Delay with $h_a = h^{(2)}$ and $P_B = 10^{-7}$

examples in section 5. However, the computational complexity of the DP approach increases exponentially as the buffer size increases. In this section, we use the imbedded Markov chain technique[2] to analyze the $M|D^B|1|L$ queueing model and provide a much easier method to evaluate P_B and $\bar{P}$.

Let $t_0^d, t_1^d, \cdots, t_n^d, \cdots$ represent the ending instants of every busy block as shown in Figure 4. In fact the time sequence $\{t_n^d\}$ denotes the sequence of the customer (packet) departure instants. Let Q_n be the queue length at t_n^d, i.e., the number of buffered packets right after the n-th block of transmission. Thus, the sequence $\{Q_n\}$ forms a Markov chain. Let ω_x^d denote the steady state probability of this Markov chain. Note that ω_x^d is probability that there are x packets in the buffer at the departure points t_n^d. The probabilities ω_x^d, for $x = 0, 1, \cdots, L$ can be easily obtained using the imbedded Markov chain technique [16] which is given in Appendix C. Since for a bulk serving system the probabilities ω_x^d at the departing points is not equivalent to the steady state probability seen by an outside observer when $r_a > 1$ [17], we cannot use ω_x^d to compute P_B and $\bar{P}$ directly via equations (9) and (11). Hence, we consider an alternative way to calculate P_B and $\bar{P}$ by exploiting the probabilities ω_x^d as will be shown now.

Let R_n denote the number of packets transmitted in block $[t_n^d - \triangle t, t_n^d)$. Assume the queue length at

[2] In reference [14], another technique, the supplementary variable technique is employed by which the closed form of the steady state probability $\omega(\mathbf{v})$ may be derived. However, the complexity of this technique makes it not so attractive for practical system design.

t_{n-1}^d is Q_{n-1} and F_{n-1} packets arrive during the vacation between two busy block, i.e, $[t_{n-1}^d, t_n^d - \triangle t)$. Then, if given Q_{n-1} and F_{n-1}, we have

$$R_n = \min\{r_a, Q_{n-1} + F_{n-1}\} \tag{31}$$

Note $R_n \in \mathcal{X} = \{0, 1, \cdots, L\}$. Consider a communication window of N blocks, and assume there are N_d busy blocks out of these N blocks. Thus the average number of transmitted packets in the busy period is defined as

$$\bar{R} \triangleq \lim_{N_d \to \infty} \frac{1}{N_d} \sum_{n=0}^{N_d} R_n \tag{32}$$

Due to the ergodicity of the Markov chain $\{Q_n\}$, $\bar{R}$ can be also obtained by taking expectations as

$$\begin{aligned}
\bar{R} &= \sum_{R_n \in \mathcal{X}} R_n \cdot \Pr\{R_n\} \\
&= \sum_{R_n \in \mathcal{X}} \sum_{Q_{n-1}} \sum_{F_{n-1}} R_n \cdot \Pr\{R_n \,|\, Q_{n-1}, F_{n-1}\} \Pr\{Q_{n-1}\} \Pr\{F_{n-1}\} \\
&= \sum_{x=0}^{L} \sum_{j=0}^{\infty} \min\{r_a, x + j\} \cdot \omega_x^d \cdot f_j
\end{aligned} \tag{33}$$

where f_j is defined in (22) and ω_x^d is the steady state probability of the queue length at the departing instants which we have obtained. Note that the summation $\sum_{n=0}^{N_d} R_n$ in (32) is also the total number of transmitted packets in the communication window N. Therefore, the packet block rate is given as

$$P_B = 1 - \lim_{N \to \infty} \left\{ \frac{1}{\lambda N} \sum_{n=0}^{N_d} R_n \right\} = 1 - \frac{\bar{R}}{\lambda} \cdot \lim_{N \to \infty} \frac{N_d}{N}$$

where the fraction $\frac{N_d}{N}$ is the fraction of time the system is busy. Since the system is busy unless it is in a vacation or the buffer is empty, in Appendix D, we derive the system busy ratio to be

$$\lim_{N \to \infty} \frac{N_d}{N} = \frac{\alpha}{1 + \frac{\omega_0^d f_0}{1 - f_0 v_0}}, \tag{34}$$

where ω_0^d is the probability of an empty buffer at the departure instants. f_0 and v_0 denote the probabilities of no packet arriving in the vacation time (23) and in one block (19) respectively. As numerical examples, Figure 7 shows the packet block rate calculated from equation (34) with given control parameters (r_a, h_a and L). In the figure, we also depict the packet block rate obtained by the DP approach

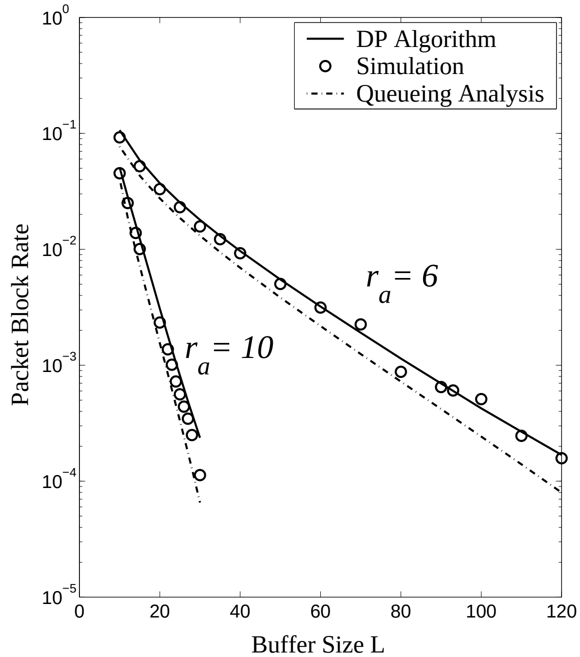


Figure 7: Packet Block Rate with $h_a = h^{(2)}$

and by the Monte Carlo simulation. It is observed that equation (34) works well to estimate the packet loss rate of the simplified policy.

For computing the average transmit power, let us first define

$$P^d(R_n) \triangleq \sum_{\substack{s\in\mathcal{H}\\ h>h_a}} q_s P_m(s, R_n) \tag{35}$$

where s is the channel state and q_s is the steady state probability of channel state in (3). $P^d(R_n)$ gives the average transmit power for transmitting R_n packets in one block. Then the long term average transmit power is given as

$$\begin{aligned} \bar{P} &= \lim_{N\to\infty} \frac{1}{N}\sum_{n=0}^{N_d} P^d(R_n) = \lim_{N\to\infty} \frac{N_d}{N}\cdot \sum_{R^d\in\mathcal{X}} P^d(R_n)\cdot \Pr\{R_n\} \\ &= \frac{\alpha}{1+\frac{\omega_0^d f_0}{1-v_0 f_0}} \cdot \left\{ \sum_{x=0}^{L}\sum_{j=0}^{\infty} P^d(\min\{r_a, x+j\})\cdot \omega_x^d \cdot f_j \right\} \end{aligned} \tag{36}$$

Since the average transmit power $\bar{P}$ is non-increasing in the average delay constraint D_{avg} for a given packet loss rate, the maximum average transmit power is reached when the average delay is minimum (30). This corresponds to the situation that in every busy block, the transmitter transmits all of

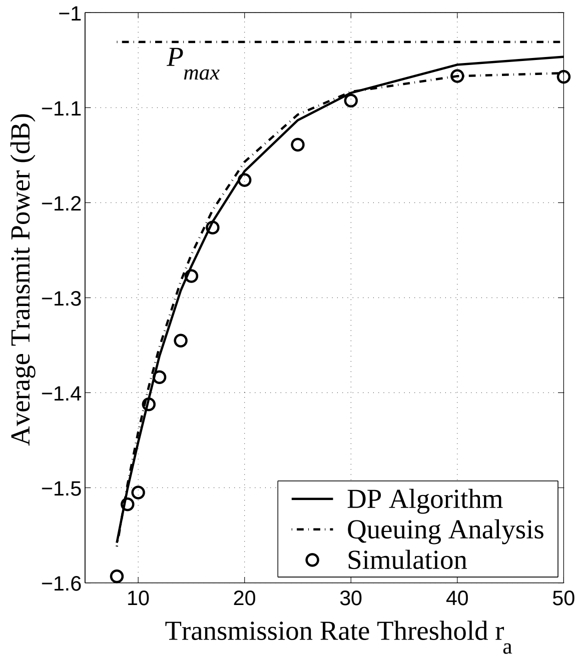


Figure 8: Average Transmit Power with $h_a = h^{(2)}$

the buffered packets. In other words, the transmitted packets in each block are the packets that arrive within the interval between the two busy blocks. Hence, an upper bound for the average transmit power is given as

$$\bar{P}_{max} = \lim_{r_a \to \infty} \bar{P} = \frac{1}{1 - v_0 f_0} \left[\sum_{i=1}^{\infty} \sum_{j=0}^{\infty} v_i f_j P^d(i+j) + v_0 \sum_{j=1}^{\infty} f_j P^d(j) \right] \tag{37}$$

where v_i and f_j are given in (19) and (22), respectively.

Figure 8 shows the packet average transmit power obtained by queueing analysis when $h_a = h^{(2)}$, where we also depict the average power obtained by the DP approach and by the Monte Carlo simulation. The results show that the queueing analysis provides a good match to both the DP and simulation results.

In summary, considering a transmission system using the simplified scheduling policy (which is determined through three control parameters, namely a transmission rate threshold r_a, a channel state threshold h_a and a buffer size L), the QoS metrics (i.e., average transmit power $\bar{P}$, average buffered delay $\bar{D}$ and packet loss rate P_B) can be approximated by equations (36), (28) and (34) respectively. Therefore, with the delay and packet constraints D_{avg} and η, the suboptimal policy can be derived by minimizing the average transmit power.

7 Conclusion

In this paper, a simplified scheduling scheme was proposed for packet data communications over fading channels. Firstly we found the optimal policy that minimized the average transmit power under constraints on both average delay and packet loss. This problem was the dual problem of the problem solved by Rajan et. al. [1] where the packet loss rate was minimized with constraints on average delay and power. Since a complicated table-look-up was needed under the optimal policy in the implementation, we designed a very simple transmission policy determined by three parameters: a transmission rate threshold, a fading channel state threshold and the transmission buffer size. It was shown that the minimum average power with the simplified scheme was very close to that achieved by the optimal policy. Under a relaxed packet loss constraint, we can reduce the buffer size by using the simplified policy. Further, we modeled the transmission system with the simplified policy as a single server bulk service queue and derived analytical approximations to the QoS metrics that are precise enough for policy design.

A Proof of Theorem 2

Proof: The proofs of statement 1 and 2 are similar to the proofs of Lemma 5.2.4 and Lemma 5.2.6 in reference [11] and are not given here.

Firstly, we assume the capacity of the buffer is big enough so that no overflow occurs. Thus, packet loss rate is reduced to the packet dropping rate. Then we have the following claim.

Claim 1 *The solution of the following problem is also the solution of Problem A:*

$$\bar{P}_{min} = \max_{\beta_1,\beta_2} \Big\{ J^*(\beta_1,\beta_2) - \beta_1 D_{avg} - \beta_2 \eta \Big\} \tag{38}$$

for $\beta_1, \beta_2 \in [0,\infty)$*, and*

$$J^*(\beta_1,\beta_2) = \lim_{\epsilon \to 1}(1-\epsilon) J^*_\epsilon(\mathbf{v} \,|\, \beta_1,\beta_2), \quad \forall \mathbf{v} \in \mathcal{S} \tag{39}$$

where $J^*_\epsilon(\mathbf{v} \,|\, \beta_1,\beta_2)$ *satisfies the following equality*

$$\begin{aligned} J^*_\epsilon(\mathbf{v} \,|\, \beta_1,\beta_2) \;&=\; \min_{u,r} \Big\{ P_m(s, r(\mathbf{v})) + \beta_1 x + \beta_2 (u(\mathbf{v}) - r(\mathbf{v})) \\ &\quad + \epsilon \sum_{s',\theta} p_{ss'} v_\theta J^*_\epsilon(f(x-u(\mathbf{v}),\theta), s \,|\, \beta_1,\beta_2) \Big\} \end{aligned} \tag{40}$$

with $p_{ss'}$ the transition probability between channel states, v_θ as given in (19) and function $f(x-u(\mathbf{v}))$ as given in (6).

Proof: Due to the ergodicity of the channel model, all the QoS metrics can be calculated by using a long-term average in time. Therefore, for any $\beta_1, \beta_2 \in [0, \infty)$, consider the following long-term average weighted combination

$$L(\beta_1, \beta_2) = \limsup_{N \to \infty} \frac{1}{N} \mathbf{E} \left\{ \sum_{i=1}^{N} \left[P_i + \beta_1 x_i + \beta_2 (U_i - R_i) \right] \right\} \tag{41}$$

The problem of minimizing (41) over all policies π is a MDP. The constants β_1 and β_2 can be interpreted as Lagrange multipliers associated with the constraints. Since the channel states form a Markov chain where all the states communicate, in the above finite state average cost problem, the minimum value of (41) does not depend on the initial state [12]. Thus, we define the minimum average cost as follows:

$$J^*(\beta_1, \beta_2) \stackrel{\triangle}{=} J(\mathbf{v} \,|\, \beta_1, \beta_2) = \min_{\pi} L(\beta_1, \beta_2) \tag{42}$$

where $\mathbf{v}$ is any initial state. Taking into account the constraints on average delay and packet loss, the problem given in (38) can be treated as equivalent to Problem A so long as $(P_i + \beta_1 x_i + \beta_2 (U_i - R_i))$ is a convex function of the policy π [13]. The convexity is guaranteed by the choice of the power function as given in (4). This proves claim 1. $\square$

Note that if the no buffer overflow assumption does not hold, then an additional constraint explicitly characterizing buffer overflow can be included to prove a more general version of Claim 1.

For any state $\mathbf{v}$, equation (40) (also referred to as the optimality equation) can be decoupled into a summation of three terms as:

$$\begin{aligned} J^*_\epsilon(\mathbf{v} = (x, s)) \;&=\; \beta_1 x + \min_r \left\{ P_m(s, r(\mathbf{v})) - \beta_2 r(\mathbf{v}) \right\} \\ &\quad + \min_u \left\{ \beta_2 u(\mathbf{v}) + \epsilon \sum_{s', \theta} p_{ss'} v_\theta J^*_\epsilon(f(x - u(\mathbf{v}), \theta), s) \right\} \end{aligned} \tag{43}$$

Note that the optimal transmission rate scheme r^* and withdrawal scheme u^* are determined by the second term and the third term of the right hand side respectively. Since by design, $r(\mathbf{v}) \le u(\mathbf{v})$ has to be satisfied for all states $\mathbf{v}$, we consider the following constrained optimization problem:

$$\min_r \left\{ P_m(s, r(\mathbf{v})) - \beta_2 r(\mathbf{v}) \right\} \quad \text{subject to} \quad r(\mathbf{v}) \le u(\mathbf{v}) \tag{44}$$

It is easy to verify that the solution to the above problem satisfies equation (15). Further at the solution, the threshold rate $R_u(s,\eta)$ is given as

$$R_u(s,\eta) = \left[\frac{S_b}{2S_p} \log_2 \frac{\beta_2^* h^{(k)} S_b}{2(\ln 2) S_p} \right] \tag{45}$$

where $[x]$ denotes the nearest integer to x and β_2^* is solution to the maximization problem in (38). This concludes the proof of Theorem 2. $\square$

B The Average Residual Service Time

For the approximated queueing system in section 6.1, the mean residual time W is comprised of three parts: mean residual service time (denoted by W_1), mean residual idle time (W_2) and mean residual vacation time (W_3). During the busy block, the system is modeled as a $M|D|1|L$ queue. When L is enough big, from the P-K formula, $W_1 \approx \frac{\lambda}{2r_a^2}$. For the mean residual idle time, $W_2 = (\frac{1}{2} + \frac{1-\alpha}{\alpha})(1-\rho)$, where $1-\rho$ is the probability that the approximated system is not busy for a system load ρ. For the mean residual vacation time, $W_3 = (\frac{1}{2} + \frac{1-\alpha}{\alpha})(1-\alpha)\rho$, where $(1-\alpha)\rho$ is the probability that the system is on a mandatory vacation with a non-empty buffer. Thus equation (29) follows. $\square$

C Steady State Probabilities ω_x^d for the Markov Chain $\{Q_n\}$

Consider the Markov chain $\{Q_n\}$ where Q_n is the number of packets in the buffer at the n-th departure instant t_n^d. The transition probability matrix of $\{Q_n\}$ is denoted by $\mathbf{T}_{L\times L} = \{p_{ij}\}$ where $p_{ij} = \Pr\{Q_n = j \,|\, Q_{n-1} = i\}$ for $i,j = 0,1,\cdots,L$. Assume F_{n-1} are the number of packets that have arrived during the vacation $[t_{n-1}^d, t_n^d - \triangle t)$ and V_{n-1} is the number of packets that have arrived during the busy block $[t_n^d - \triangle t, t_n^d)$. The distribution of F_{n-1} and V_{n-1} is given in (22) and (19), respectively. Then we have the following recursive relationship.

$$Q_n = \begin{cases} Q_{n-1} + F_{n-1} - r_a + V_{n-1} & Q_{n-1} + F_{n-1} \geq r_a \\ V_{n-1} & 0 < Q_{n-1} + F_{n-1} < r_a \end{cases} \tag{46}$$

If $Q_{n-1} + F_{n-1} = 0$, the system will experience one block of idle state, and Q_n will depend on the number of packets that have arrived in this idle interval. As a result, for $j = 0,1,\cdots,L-1$, the transition probability can be expressed as

$$p_{ij} = \begin{cases} \dfrac{f_0 \phi}{1 - v_0 f_0} + \displaystyle\sum_{k=1}^{r_a} f_k v_j + \sum_{k=r_a+1}^{\infty} f_k v_{j+r_a-k}, & i = 0; \\ \displaystyle\sum_{k=0}^{r_a - i} f_k v_j + \sum_{k=r_a-i+1}^{\infty} f_k v_{j-i+r_a-k}, & r_a \geq i > 0; \\ \displaystyle\sum_{k=0}^{\infty} f_k v_{j-i+r_a-k}, & i > r_a. \end{cases} \tag{47}$$

where

$$\phi = v_j \left\{ f_0 \sum_{k=1}^{r_a} v_k + v_0 \sum_{k=1}^{r_a} f_k + \sum_{k=1}^{r_a-1} \sum_{l=1}^{r_a-k} f_k v_l \right\} + \sum_{k=1}^{j} \sum_{l=0}^{r_a+k} v_l f_{r_a+k-l} v_{j-k} \tag{48}$$

When $j = L$, it follows that

$$p_{iL} = \Pr\{Q_{n+1} = L \,|\, Q_n = i\} = 1 - \sum_{j=0}^{L-1} p_{i,j} \tag{49}$$

The steady distribution $\Omega_x^d = \{\omega_x^d\}$ is determined from the set of equations

$$\Omega_x^d \mathbf{T} = \Omega_x^d \quad \text{and} \quad \sum_{x=0}^{L} \omega_x^d = 1$$

From the above transition probabilities, any standard numerical procedure can be used to derive the steady state probabilities ω_x^d of $\{Q_n\}$. Note that the computational complexity of solving for ω_x^d increases as the buffer size L increases. However, it is still substantially smaller when compared to traditional value iteration DP approaches for solving Problem B.

D System Busy Ratio

To derive equation (34), consider a communication window N and let N_d denote the number of busy blocks. Let N_I denote the number of idle blocks and N_0 denote the number of departure points with empty buffer. Since the probability of being in vacation is α, we have

$$N_d + N_I = (1 - \alpha) N \tag{50}$$

If at a departing instant, the buffer is empty, the average number of the idle blocks following this departing instant is given as $\frac{f_0}{1-v_0 f_0}$. Hence, it follows that

$$N_I = \frac{N_0}{1 - v_0 f_0} \tag{51}$$

From (50) and (51), equation (34) follows, where $w_0^d = \frac{N_0}{N_d}$. □

References

[1] D. Rajan, A. Sabharwal, and B. Aazhang, "Transmission Policies for Bursty Traffic Sources on Wireless Channels," 35th Annual Conference on Information Sciences and Systems, Baltimore, Mar. 2001.

[2] A. P. Chandrakasan and R. W. Brodersen, "Minimizing power consumption in digital CMOS circuits," IEEE Proc., Vol. 32, No. 4, Apr. 1995.

[3] A. Chockalingam, M. Zorzi, "Energy Efficiency of Media Access Protocols for Mobile Data Networks," IEEE Trans. on Comm. Vol. 46, No.11, pp. 1418-1421, Nov. 1998.

[4] H. Woesner, J.-P. Ebert, M. Schlager, and A. Wolisz, "Power-saving mechanisms in emerging standards for wireless LANs: the MAC level perspective", IEEE Personal Communications Vol. 5 No. 3, pp. 40-48, June 1998.

[5] Q. Zhang, W. Zhu, Zu Ji, and Y. Zhang, "A Power-Optimized Joint Source Channel Coding for Scalable Video Streaming over Wireless Channel", IEEE International Symposium on Circuits and Systems (ISCAS) 2001, May, 2001, Sydney, Australia.

[6] R. Koenen, F. Pereira, and L. Chiariglione, "MPEG-4: Context and Objectives," Image Communication Journal: MPEG-4 Special Issue, Vol. 9, No. 4, May 1997.

[7] B. E. Collins and R. L. Cruz, "Transmission Policies for Time Varying Channels with Average Delay Constraints" Proc. 1999 Allerton Conf. on Communication Control, & Comp., Monticello, IL.

[8] D. Rajan, A. Sabharwal, and B. Aazhang, "Delay and Rate Constrained Transmission Policies over Wireless Channels," Proc. Globecom 2001, San Antonio, November 2001.

[9] A. Sabharwal, D. Rajan and B. Aazhang, “Dual problems in power control,” 39th Allerton Conf. on Communication, Control and Computing., (Allerton, IL), October 2001.

[10] R. A. Berry and R. G. Gallager, “Communication over Fading Channels with Delay Constraints,” IEEE Trans. on Information Theory, Vol. 48, No. 5, pp. 1135-1149, May 2002.

[11] R. A. Berry, *Power and Delay Trade-offs in Fading Channels*. Ph.D. Thesis, Massachusetts Institute of Technology, June, 2000.

[12] M. L. Puterman, *Markov Decision Processes : Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 1994.

[13] Eitan Altman, *Constrained Markov Decision Processes*. Chapman & Hall/CRC.1999

[14] M. L. Chaudhry and J. G. C. Templeton, *A First Course in Bulk Queues*. John Wiley & Sons, 1983.

[15] D. Bertsekas, R. Gallager, *Data Networks,* Prentice-Hall, 1992

[16] L. Kleinrock, *Queuing Systems*. John Wiley & Sons, 1975.

[17] R. B. Cooper, *Introduction to Queuing Theory, 2nd Edition*. Elsevier North Holland, Inc. 1981